HAL
open science

Mosaïquage de données géoréférencées du cadastre ancien : vers une approche basée sur les graphes

Jean-Marc Beveraggi

To cite this version:

Jean-Marc Beveraggi. Mosaïquage de données géoréférencées du cadastre ancien : vers une approche basée sur les graphes. Sciences de l'ingénieur [physics]. 2018. dumas-02092715

HAL Id: dumas-02092715

https://dumas.ccsd.cnrs.fr/dumas-02092715v1

Submitted on 8 Apr 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

CONSERVATOIRE NATIONAL DES ARTS ET METIERS

ECOLE SUPERIEURE DES GEOMETRES ET TOPOGRAPHES

MEMOIRE

présenté en vue d'obtenir

le DIPLOME D'INGENIEUR CNAM

SPECIALITE : Géomètre et Topographe

par

Jean-Marc BEVERAGGI

Mosaïquage de données géoréférencées du cadastre ancien – vers une approche basée sur les graphes

Soutenu le 03 Juillet 2018

JURY

PRESIDENT : M. PROUDHOM Christophe

MEMBRES :

M. FOLLIN Jean-Michel	**Maître de stage**
Mme. FOURNIER Marie	**Examinateur**
Mme. SIMONETTO Elisabeth	**Maître de stage**
Mme. VERDUN Jérôme	**Professeur référent**

Remerciements

En premier lieu, je tiens à remercier mes deux maitres de stages, Mme Elisabeth SIMONETTO et M. Jean-Michel FOLLIN pour leur aide, leur l'accompagnement sans faille, ainsi que pour le temps qu'ils m'ont consacré tout au long de ce TFE.

Par la suite je tiens à remercier mes deux professeurs référents, Mme Nathalie THOMMERET pour son suivi dans la première partie de ce TFE et pour ses remarques avisées quant à la rédaction de mon mémoire, et M. Jérôme VERDUN pour son accompagnement dans la seconde partie de mon TFE pour les corrections apportées au mémoire dans sa globalité.

Je remercie également M. Kamaldeep SINGH OBEROI pour les connaissances qu'il a pu nous apporter sur la théorie des graphes lors de sa visite à l'ESGT.

Je remercie aussi M. Fréderic DURAND d'avoir pris de son temps pour m'aider dans l'utilisation du serveur de calculs de l'école.

Je remercie mes collègues de la T0 pour la convivialité et l'aide dont ils ont fait preuve à mon égard tout au long de ce TFE.

Enfin je tiens tout particulièrement à remercier mes parents, sans qui je n'aurais jamais pu poursuivre mes études, pour leur soutien moral et financier.

Glossaire

La **vectorisation** est le procédé permettant de transformer une image matricielle du format raster (ici papier scanné) au format vecteur (points, lignes, surfaces).

Le **géoréférencement**, attribution d'une géoréférence à un objet, est l'opération qui consiste à passer d'une localisation relative des entités géographiques à leur localisation absolue dans un système de coordonnées géographiques reconnu. (d'après le Glossaire des SIG [en ligne] http://www.emse.fr/tice/uved/SIG/Glossaire/co/Georeference.html)

Le **mosaïquage** (ou assemblage) est l'action de faire coïncider plusieurs entités entre elles, après leur géoréférencement, tout en respectant les règles de topologie qui les unissent.

Le **cadastre** est un outil foncier permettant le calcul de la taxe foncière. Il a été instauré par une loi du 15 septembre 1807 par Napoléon 1er. Il a subi plusieurs modifications (rénovation...). Il se subdivise en plusieurs niveaux d'échelle (de la plus générale à la plus précise) : le tableau d'assemblage, la section et la feuille. (Pour plus d'informations sur le cadastre et son histoire, se référer aux TFE précédents (Odie 2017 et Fahrasmane 2016))

Abréviations

TFE : Travail de Fin d'Etudes

SIG : Système d'Information Géographique

EMQ : Erreur Moyenne Quadratique.

PCI : Plan Cadastrale Informatisé

Table des matières

REMERCIEMENTS **2**

GLOSSAIRE **3**

ABREVIATIONS **3**

TABLE DES MATIERES **4**

INTRODUCTION **6**

I CONTEXTE ET DONNEES **8**

I.1 LA CHAINE DE TRAITEMENT ELABOREE PAR LE LABORATOIRE GEF 8

I.2 LES DONNEES 10

I.2.1 Aspects généraux 10

I.2.2 Les données utilisées 12

I.3 ANALYSE DES OUTILS EXISTANTS 12

II ASPECTS THEORIQUES **17**

II.1 TOPOLOGIE 17

II.1.1 Les relations topologiques 17

II.1.2 Les outils de correction topologique 17

II.2 LA THEORIE DES GRAPHES 19

II.2.1 Histoire des graphes 19

II.2.2 Définition 20

II.2.3 Projets utilisant la théorie des graphes 20

II.3 OUTILS D'ANALYSE 22

III METHODOLOGIE **24**

III.1 NOTIONS DE RELATIONS ENTRE LES DIFFERENTES ENTITES 24

III.2 MOSAÏQUAGE 27

III.2.1 Présentation du traitement 27

III.2.2 Suppression des chevauchements 29

III.2.3 Suppression des trous 31

III.3 APPLICATION DE LA THEORIE DES GRAPHES 33

III.3.1 Différents types de graphes 33

III.3.2 Traitement d'une planche 36

III.3.3 Graphe de plusieurs planches 38

IV PRESENTATION DES RESULTATS 43

IV.1 LES RESULTATS DU MOSAÏQUAGE 43

IV.1.1 Année 1813 43

IV.1.2 Année 1850 44

IV.1.3 Années 1972-1974 45

IV.2 LES RESULTATS DES GRAPHES 46

IV.2.1 Année 1813 46

IV.2.2 Année 1850 46

IV.2.3 Année 1973 47

IV.3 LIENS ENTRE MOSAÏQUAGE ET THEORIE DES GRAPHES 47

CONCLUSION ET PERSPECTIVES 50

BIBLIOGRAPHIE 52

TABLE DES FIGURES 54

LISTE DES TABLEAUX 55

Introduction

Le cadastre fait très certainement partie des documents les plus importants de l'histoire de la France, il est utilisé pour le calcul de la taxe foncière depuis 1813 sous le régime de Napoléon 1er. Depuis peu, il y a eu un engouement pour l'étude des documents anciens tels que les cartes, les plans terriers et compoix avec le projet Architerre, et les anciens plans cadastraux. C'est pourquoi le Laboratoire Géomatique et Foncier du Conservatoire National des Arts et Métiers en a profité pour lancer un projet nommé « le cadastre comme outil d'analyse de l'évolution des territoires ». Afin d'étudier l'évolution de ces territoires, nous disposons de différentes planches de cadastre de deux communes situées au sud de la Sarthe : Aubigné-Racan et Vaas. Ces planches de cadastre couvrent les années 1813, 1850 et 1972-74. Afin d'analyser ces données, qui sont sous forme raster, il va nous falloir les vectoriser et les géoréférencer. Seulement, l'opération de géoréférencement ne suffit pas pour analyser correctement l'évolution du parcellaire à l'échelle de plusieurs sections. En effet, à cause des erreurs dues à la conservation des planches, de la vectorisation semi-automatique puis du géoréférencement, les planches ne vont pas pouvoir s'agencer correctement.

C'est pourquoi le mosaïquage des sections vectorisées nécessite de corriger les erreurs topologiques entre ces dernières et de prendre en compte les routes et cours d'eau qui peuvent séparer celles-ci. L'objectif de ce travail de fin d'études sera donc de poursuivre la mise en place de la chaîne de traitement semi automatisée, dont les parties vectorisation et géoréférencement ont déjà été réalisées. Ce TFE s'intéressera tout particulièrement à l'aspect mosaïquage de la chaîne. Grâce à cette chaîne de traitement, nous serons en mesure d'établir une base de données multi-époques et d'analyser les évolutions d'un territoire (division, fusion, remembrement, opération d'aménagement…) sur une période donnée.

Il existe des outils pour corriger les erreurs topologiques, disponibles sur des logiciels propriétaires tels qu'ArcGIS, mais la chaîne de traitement proposée par le GeF s'appuie sur des outils Open Source tels que le langage Python ou encore les logiciels QGIS et GRASS, ce dernier proposant de nombreux outils topologiques. Dans ce TFE nous proposerons une chaine de traitement semi-automatisée composée en grande partie d'outils QGIS, permettant de corriger les soucis de chevauchements et de trous entre planches. Nous proposerons également des réflexions méthodologiques pour développer une approche de mosaïquage basée sur la théorie des graphes. Notre idée est d'utiliser les relations de voisinage à l'échelle des parcelles

et à celle des sections afin de guider les corrections à apporter sur les données lors du mosaïquage.

Concernant la méthode basée sur la théorie des graphes, nous emploierons le langage de programmation Python, qui utilise une syntaxe simple et épurée ce qui permet de gagner du temps dans la programmation des scripts.

Afin de répondre au problème posé, nous vectoriserons trois jeux de données, correspondant aux trois années disponibles, eux-mêmes composés de cinq planches, pour la commune d'Aubigné-Racan. Il est à noter que les graphes seront issus d'entités qui ont été vectorisées semi automatiquement à partir de document plus ou moins bien conservés ce qui influe sur la qualité de la vectorisation. Nous pourrons donc trouver des aberrations, telles que des parcelles manquantes, des micro-polygones ou bien des erreurs de géométrie. Tout cela influera sur la forme et l'exactitude du graphe généré.

Notre problématique est la suivante : Quelle est la meilleure méthode de mosaïquage s'appliquant à notre étude ?

Dans une première partie, nous développerons le contexte global dans lequel s'inscrit ce TFE ainsi que les travaux déjà effectués concernant la chaîne de traitement proposée par le GeF. Nous décrirons ensuite les données utilisées ainsi que les différents outils mis à dispositions dans les précédents TFE.

Dans une seconde partie nous étudierons les aspects théoriques de notre travail avec des notions sur la théorie des graphes et des notions portant sur la topologie.

Dans une troisième partie nous mettrons en place une méthodologie afin de proposer un mosaïquage des données.

Dans une quatrième partie nous mettrons en application la méthodologie, proposée dans la troisième partie, avec une présentation et une critique des résultats, pour enfin, mettre en relation la théorie des graphes et le mosaïquage.

I. Contexte et données

Ce TFE fait suite à un projet pré-professionel effectué en 2015, et de deux autres TFE, celui de Maïté Fahrasmane (2016) et celui de Charlotte Odie (2017). Il s'inscrit dans le projet « Le cadastre comme outil d'analyse de l'évolution des territoires » du laboratoire Géomatique et foncier (GeF) du CNAM. Le but de ce projet est de produire une chaîne de traitement semi-automatique permettant de vectoriser des planches de cadastre ancien, de les géoréférencer puis de les intégrer dans une base de données multi-époques. Cela servira à qualifier les évolutions du parcellaire (division, fusion...) à différentes échelles et d'étudier les transformations du territoire

I.1 La chaine de traitement élaborée par le laboratoire GeF

La chaîne de traitement proposée par l'équipe L2G (Géodésie et Géomatique) du laboratoire GeF se veut être automatisée au maximum, reproductible et elle permettra au final d'analyser l'évolution du territoire. L'automatisation permettant une plus grande rapidité et la suppression des erreurs d'opérateurs. La chaîne est décrite dans la figure 1 ci-après. Les étapes principales sont la **vectorisation, le géoréférencement, le mosaïquage et l'analyse temporelle à l'aide d'une base de données vectorielles multi-dates.**

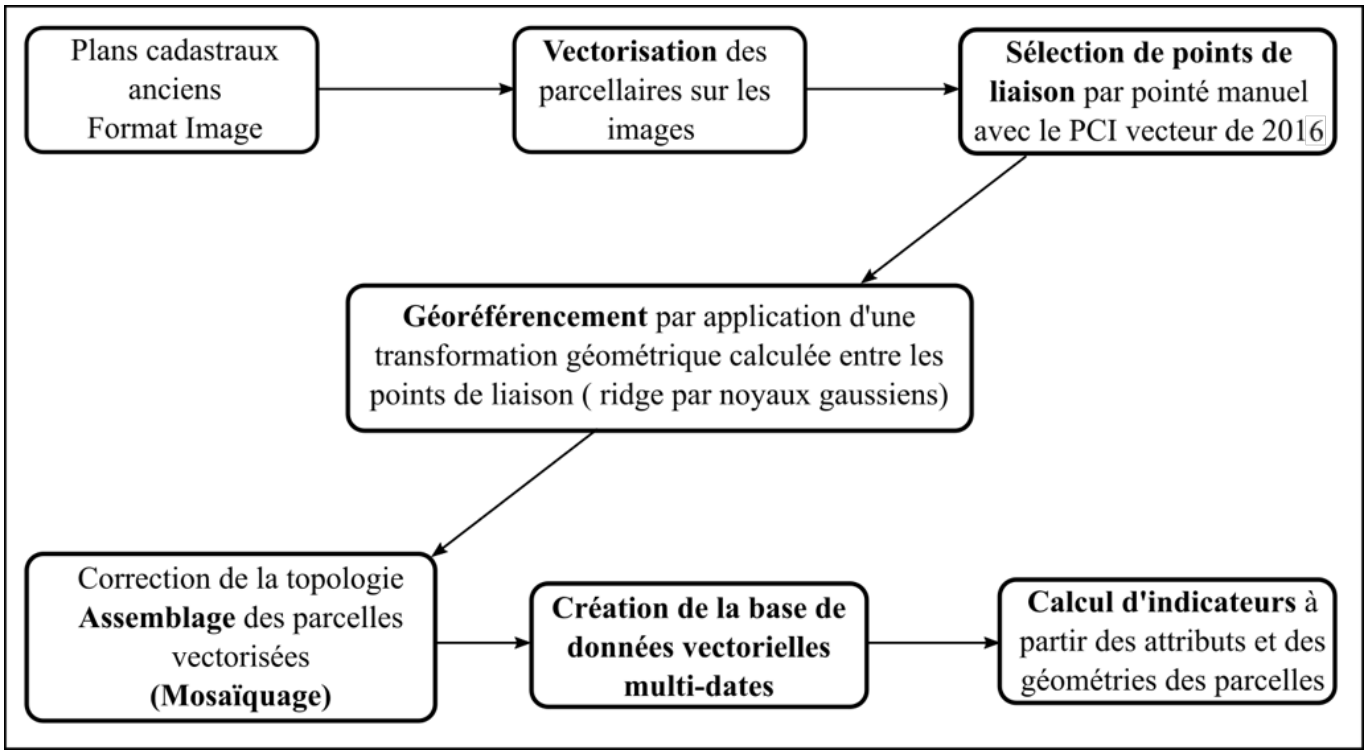


Figure 1 Principe d'enchaînement des étapes de la chaine méthodologique du Laboratoire GeF, d'après Fahrasmane (2016)

Ces travaux font suite à un projet pré-professionnel réalisé par des élèves de l'ESGT en 2015 (Labrousse et al.) qui avaient pour mission de suivre l'évolution du parcellaire sur deux communes du sud de la Sarthe : Vaas et Aubigné-Racan. Pour mener à bien le projet, les étudiants ont utilisé des planches cadastrales de différentes époques (1813, 1850 et 1972-74) ainsi que le PCI (Plan Cadastrale Informatisé) vecteur de 2012. Les planches de cadastre achetées aux archives départementales sous forme de scans, ont été vectorisées manuellement. Elles ont ensuite été géoréférencées pour être intégrées dans une base de données multi-époques, le tout dans le but de pouvoir suivre, étudier et comprendre l'évolution du parcellaire. Afin d'améliorer la chaîne de traitement, un sujet de travail de fin d'études a vu le jour.

Maïté Fahrasmane (2016) s'est penchée sur l'aspect géoréférencement de la chaîne de traitement. Plusieurs fonctions mathématiques ont été testées :

- des fonctions globales : polynomiales de degré 1, 2, 3, recalage Helmert, fonction spline à plaque mince TPS (Thin Plate Spline)
- une fonction locale : la triangulation de Delaunay
- une méthode innovante globale sensible localement dite méthode par noyau gaussien

Ces fonctions ont été codées à l'aide du langage de programmation Open Source Python. Afin de comparer les différentes méthodes entre elles, il a été mis en place une méthode de validation croisée. Cette méthode classique consiste à prendre successivement chaque point comme point de contrôle. On aura ainsi autant d'Erreurs Moyennes Quadratiques (EMQ) qu'il y aura de points ce qui en fait une méthode de validation plus robuste que si cette EMQ n'avait été calculée qu'une seule fois. Après comparaison des différentes fonctions, il s'est avéré que la fonction ridge par noyau gaussien était celle fournissant les meilleurs résultats.

Le TFE de Charlotte Odie (2017) avait pour objectif de proposer une méthode de vectorisation semi-automatique des planches de cadastre numérisées. Deux méthodes ayant des étapes de pré-traitement identiques (conversion en niveaux de gris, application d'un masque et détection de points de contour par seuillage, …) ont été évaluées. La première se base sur la transformée de Hough dite probabiliste. Elle utilise une image binaire en noir et blanc où chaque pixel est codé en 1 ou en 0. Les différents calculs associés à cette méthode ont été implémentés sous Python. La seconde est la méthode dite de « suivi de chemin » proposée dans le logiciel GRASS, avec son outil *r.to.vect*. Pour évaluer ces deux méthodes, leurs résultats ont été comparés à des planches de cadastre vectorisées manuellement. Il en ressort que c'est la

méthode de suivi de chemin de GRASS qui semble offrir les meilleurs résultats pour les planches anciennes. Ces méthodes restent à approfondir. En effet on observe les problèmes suivants :

- certaines lignes, matérialisant les limites de parcelles, se retrouvent coupées et les parcelles ne sont donc pas fermées,
- des artefacts constatés sur les planches de cadastre : ils peuvent être liés aux écritures (toponymie, numéro de parcelle, carroyage), ou aux traces laissées par le temps (pliures, tâches d'eau, tâches d'encre…). Ces dernières sont plus ou moins importantes suivant l'époque choisie et l'état de conservation de la planche. Ainsi, les planches de cadastre de 1974 donnent un résultat plus satisfaisant avec la méthode de vectorisation semi-automatique que les planches datant de 1813.

Dans ce TFE, nous allons donc nous attarder tout particulièrement sur la partie mosaïquage des planches vectorisées. Pour cela, nous avons envisagé différentes méthodes dont celle reposant sur la théorie des graphes. Cette théorie permettra d'étudier les relations de voisinages entre parcelles au sein même d'une planche mais également entre les planches. Nous proposerons également une chaine de traitement impliquant des outils proposés par QGIS afin de procéder au mosaïquage de nos données.

I.2 Les données

Afin de proposer une chaine de traitement, nous avons décidé de travailler sur la commune d'Aubigné-Racan uniquement. Par la suite, cette chaine de traitement pourra être appliquée à d'autres données, comme celles de la commune de Vaas.

I.2.1 Aspects généraux

Nous travaillons sur trois époques du cadastre de la commune d'Aubigné-Racan. Nous avons le cadastre napoléonien de 1813, celui de 1850 et celui de 1972. Les données de 1813 restent les plus difficiles à exploiter, le papier est extrêmement jauni par le temps et l'humidité, de nombreuses traces laissées par l'homme faussent la vectorisation automatique. Les planches de 1850 contiennent à peu de choses près les mêmes problèmes même si elles sont un peu mieux conservées. Quant aux planches numérisées de 1972-74, elles sont dans un état de conservation parfait et elles sont déjà en images binaires, noir et blanc. Nous pouvons constater les différences de conservation entre les différentes époques sur la figure 2 ci-après.

Figure 2 Extraits des cadastre de 1813, 1850, 1972-74 (de gauche à droite)

Le cadastre se décompose suivant plusieurs niveaux :

- La commune (tableau d'assemblage)
- Les sections (représentées par des lettres A, B, C...)
- Les feuilles, que nous appellerons également planches (représentées par la lettre de la section et d'un chiffre correspondant au numéro de la feuille A1, A2...)

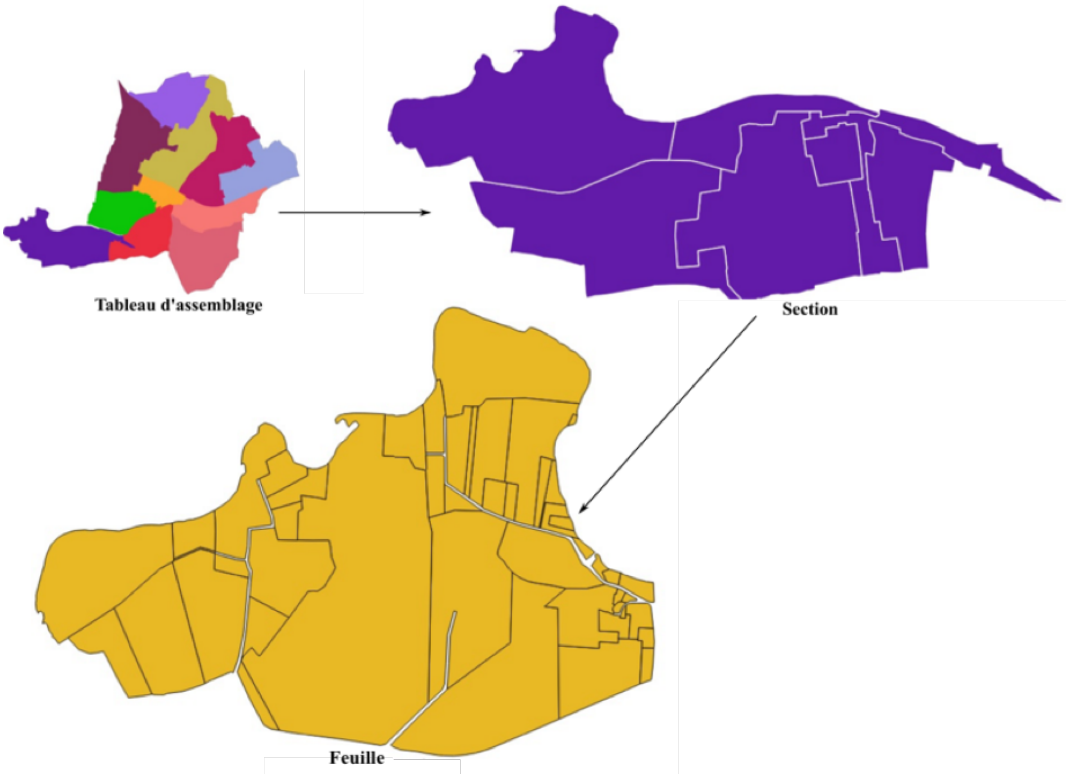


Figure 3 Différents niveaux d'échelle du cadastre, d'après Faharsmane (2016)

Année	1813	1850	1972-1974
Échelle	1/2500	1/2000	1/2000
Effectifs de planches	22	45	44
Pas du scanner	200 dpi	200 dpi	400 dpi
Taille d'un pixel dans l'image (µ)	127	127	63,5
Taille d'un pixel sur le terrain (m)	0,3175	0,254	0,127

Tableau 1 Propriétés des images scannées de la commune Aubigné-Racan, d'après Fahrasmane (2016)

I.2.2 Les données utilisées

Pour bien comparer les différentes époques, nous avons choisi de vectoriser un ensemble de cinq planches qui se superposent géographiquement sur les trois différentes époques. La superficie et la nomenclature des sections changeant d'une époque à l'autre, l'emprise des cinq planches ne sera pas exactement la même entre 1813, 1850 et 1972-74, mais nous avons fait en sorte de travailler sur la zone d'intersection des trois emprises, commune aux trois époques. Les différentes données utilisées sont listées dans le tableau 2 ci-après.

Année	Section	Image (.JPG)
1813	C1, C2, C3, F1, F2	006, 007, 008, 012, 013
1850	J1, J2, J3, K1, K2	116, 117, 118, 119, 120
1972-74	K1, K2, K3, P2, P3	153, 154, 169, 170

Tableau 2 Données utilisées suivant les différentes époques.

I.3 Analyse des outils existants

Dans cette partie nous allons référencer tous les soucis techniques rencontrés lors de l'utilisation des différents outils créés ou utilisés lors des précédents TFE. Notamment concernant les problèmes de vectorisation et de géoréférencement.

- Programme Python : Vectorisation par Transformée de Hough dite probabiliste

Plusieurs facteurs vont influer sur la vectorisation : marques de pliures, déchirures, tâches (eau, encre), traits de construction, papier jauni avec le temps. Tous ces facteurs vont nécessiter des post-traitements plus poussés, et augmenter le temps passé pour vectoriser une planche.

Un autre problème qui peut être résolu en amont de la vectorisation est la présence du carroyage. Il peut être supprimé à l'aide de masques. Seulement cela soulève un autre problème, en enlevant le carroyage, on va créer des discontinuités dans les limites de parcelles qui devront être reconnectées (voir partie GRASS en *infra*).

Tous les problèmes liés à l'état de conservation des planches de cadastre (voir *supra*) vont influencer sur la vectorisation. En effet lorsque l'on vectorise, le programme va détecter les lignes de contours à des endroits où il n'y en a pas (ex : pliures, tâches…). Pour limiter ces « fausses alarmes » nous avons restreint la vectorisation au contour de la feuille avec un masque et supprimé les petits objets dont la taille est inférieure à un certain seuil grâce au programme Python.

Un problème qui apparaît et qui est lié au fonctionnement de la transformée de Hough probabiliste, est le rendu en sortie du programme. Au lieu de ne générer qu'une seule ligne par contour de parcelle, le programme retourne une multitude de lignes (voir figure 4).

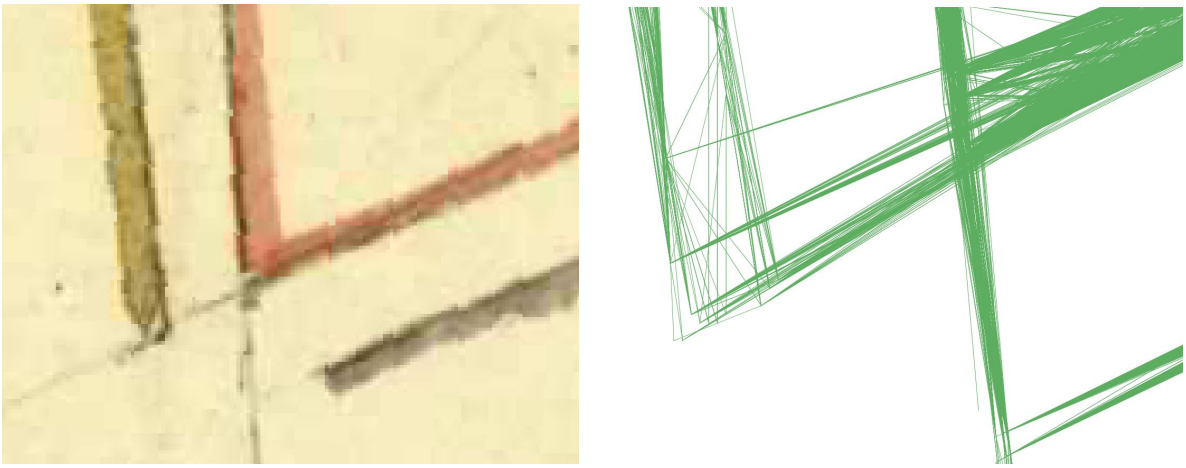


Figure 4 Problème de multi-ligne provoqué par l'épaisseur des lignes et les liserais sur le format raster

En sortie de programme, les vecteurs (au format shapefile) créés possèdent trois colonnes : ID, n° parcelle, section. Or, pour pouvoir géoréférencer ces derniers, le programme de transformation ridge par noyau Gaussien demande en entrée des données vectorielles avec

cinq colonnes dans la table attributaire (ID, n° Parcelle, Commune, Section, n° Feuille). Il ne s'agit pas réellement d'un problème, mais plutôt d'un détail d'implémentation. Ici nous aurons rajouté des champs avec des valeurs arbitraires à l'aide d'un tableur. Pour corriger ce problème il suffit tout simplement de modifier le programme de labellisation pour avoir non pas trois, mais cinq colonnes dans la table attributaire des données vectorisées.

- Outil GRASS : vectorisation par suivi de chemin :

La vectorisation sur GRASS demande en entrée une image binaire (noir ou blanc (0 ou 1)) générée ici par un programme Python utilisant la bibliothèque scikit-image. Il est possible de nettoyer cette image afin de supprimer les plus gros objets qui n'ont pas été effacés par la fonction « remove small object » de python, puis d'enlever les plus gros reliquats à l'aide d'un logiciel de retouche d'images comme Gimp. Cela permettra de faciliter le traitement effectué sous GRASS. Cette opération de nettoyage peut également être réalisée avec les outils de GRASS avant la labellisation puis la transformation en polygone de chaque parcelle, ou bien directement sous QGIS avec le format vecteur ligne des parcelles en passant la couche en édition et en supprimant les erreurs manuellement.

D'autres problèmes ont été remarqués dans cette méthode de vectorisation par suivi de chemin, notamment lors de l'étape de nettoyage (post-traitement), pour le choix des seuils de la boite à outils *v.clean* de GRASS. En effet nous pouvons nommer la fonction *snap,* qui permet de raccorder deux segments, s'ils sont à une distance inférieure à un seuil donné. Si nous lui choisissons un seuil trop haut, certaines lignes vont faire des « zig-zag », cela impacte surtout les petits éléments comme les bâtiments ou les parcelles fines ou de petite taille. Lorsque l'on utilise un seuil trop petit, les limites de parcelles coupées ne vont pas se connecter entre elles. Il faut donc trouver le bon compromis entre un seuil trop haut et un seuil trop bas.

Figure 5 Choix d'un seuil trop important pour la fonction snap : avant/après

Un souci de géométrie apparaît également à certaines intersections. Il semblerait que ce problème survienne au moment de la squelettisation de l'image. En effet, sur les planches « papier » là où les lignes s'intersectent, il y aura un trait plus épais, la squelettisation aura ainsi plus de difficultés à s'effectuer. Il s'agit ici que d'un soucis d'ordre visuel et géométrique, cela n'affecte pas la topologie de nos parcelles.

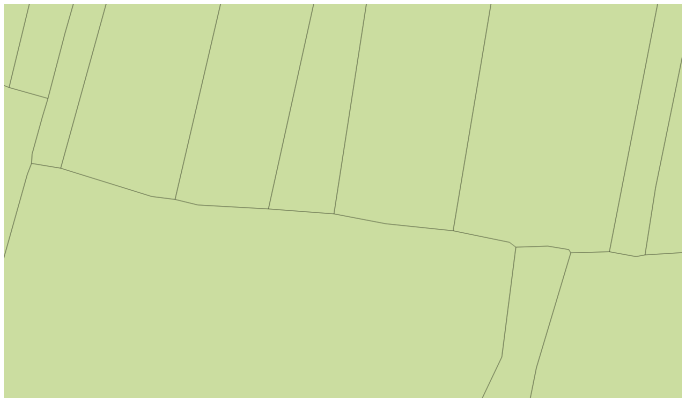


Figure 6 Croisements en "V" aux intersections de parcelles

Un autre problème a rencontré suite à la vectorisation semi-automatique par suivi de chemin. Le repère de la couche vectorisée et le repère de l'image raster ne sont pas orientés de la même façon. En effet, pour la couche vectorielle, l'axe Y est orienté vers le haut alors que pour la couche raster il est orienté vers le bas. Pour corriger cela il faut modifier la couche vecteur manuellement pour superposer le vecteur sur le raster. Si l'on ne corrige pas ce problème, nous aurons un décalage des coordonnées de la donnée géoréférencée égale à cette différence de repère. Pour solutionner ce problème nous avons utilisé le module « Affine » disponible sous QGIS pour applique une translation verticale de distance égale à la hauteur de l'image.

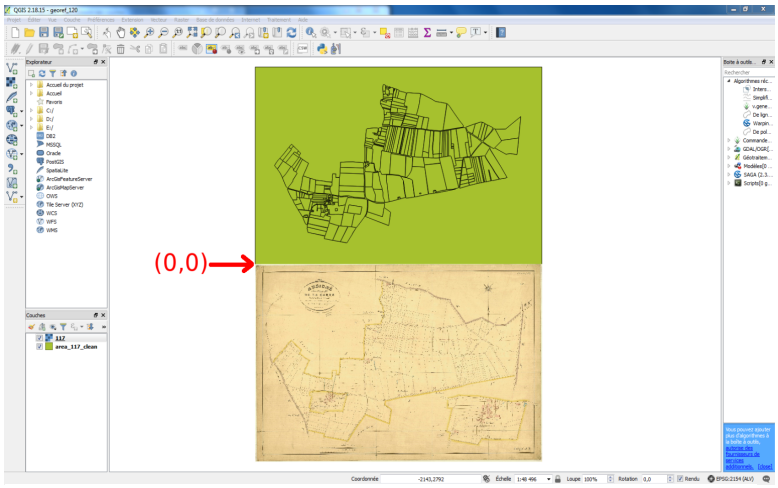


Figure 7 Différence de repères pour le vecteur et le raster

- Programme Python : Labellisation

Il peut arriver, qu'il y ait un polygone à l'intérieur d'un autre polygone (bâtiment dans une parcelle ou une parcelle enclavée dans une autre), et lors de la vectorisation de l'image labélisée, GRASS va avoir du mal à l'interpréter, et va créer le contour du polygone mais il va également continuer le contour autour du polygone qu'il contient. La seule solution que nous ayons pu mettre en place est de supprimer manuellement les nœuds en trop.

Figure 8 Problème de polygone inclut dans un autre lié à la labellisation

Les produits en sortie de la chaîne de traitement de vectorisation ne sont pas parfaits. En effet, de petits objets comme des bâtiments n'ont pas été vectorisés, des parcelles peuvent être incomplètes car elles ne sont pas cadastrées précisément (lignes en tireté non détectées) et certaines lignes de contour n'ont pas été décelées. Cela concerne tant bien les résultats du programme Python basé sur la transformée de Hough, que de l'outil de suivi de chemin de GRASS.

Enfin, les routes ne sont pas représentées, ce qui forme des trous entre les parcelles. Cela est dû à la labellisation qui nécessite des contours fermés pour créer des polygones. Or les routes sont représentées par des lignes non fermées. Il faudra donc mettre en place un post-traitement pour récupérer les linéaires des routes (partie III.3 Mosaïquage).

- Programme Python : Géoréférencement par transformation ridge par noyau Gaussien

Pour les planches les plus anciennes, sur certaines zones, le parcellaire a énormément changé. Des lotissements, des aménagements, du remembrement ont été réalisés, si bien qu'il est difficile de trouver des points d'amer pour débuter le géoréférencement. Le manque de points d'amers va réduire la qualité du géoréférencement et nous allons observer des décalages entre la donnée vectorisée et la donnée de référence.

II. Aspects théoriques

II.1 Topologie

II.1.1 Les relations topologiques

Afin de mieux comprendre le mosaïquage, il est bon de s'intéresser aux relations des différentes entités, les unes par rapport aux autres. On peut distinguer 3 types de relations spatiales : topologique, métrique et d'orientation. *Randell et al. (1992),* définit dans son article *Region Connection Calculus* huit types de relations topologiques (figure 9). Pour notre étude, nous allons nous intéresser aux relations topologiques.

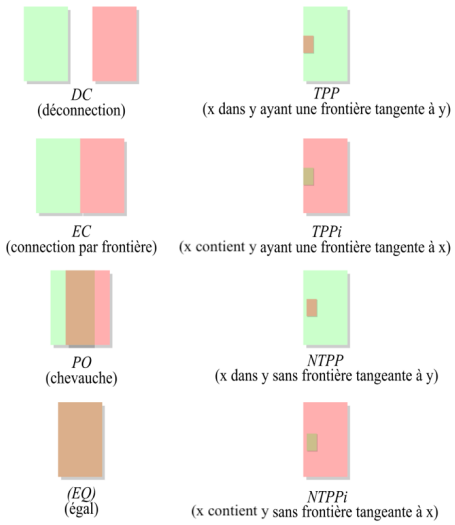


Figure 9 Relations topologiques du RCC8, d'après Randell et al. (1992)

II.1.2 Les outils de correction topologique

ArcGis est connu comme étant l'un des meilleurs logiciels permettant la manipulation de fichier de forme, mais il n'est pas le seul à proposer des solutions de corrections topologiques. En effet J. Martinez-Llario et *al* (2017) fait l'état de l'art de nombreux logiciels proposant également des outils de corrections topologiques. Il définit tout de même ArcGis comme étant « [l'outil] ayant les meilleures solutions ». L'article classifie également les différents outils suivant leurs capacités à corriger les erreurs topologiques et leur disponibilité (Open Source ou propriétaire).

Nous allons distinguer les règles topologiques qui correspondent à des critères que doivent suivre les polygones (pas de nœuds en doublons, pas de polygones superposés, pas de trous dans un polygone...), des erreurs topologiques qui sont dues au non-respect des règles topologiques et qui seront corrigées à l'aide d'outils topologiques.

Le tableau 2 nous décrit brièvement les différents logiciels qui existent en matière de correction topologiques. Pour ne citer que les plus connus, il existe le logiciel ArcGIS qui dispose de 32 règles de topologies aidant à la correction. Il existe également le logiciel GRASS que nous avons pu utiliser dans la vectorisation de nos planches avec l'outil de « suivi de chemin ». Ce logiciel est quant à lui davantage destiné à un nettoyage des données (accrochage de segments, suppressions de petites aires...). Vient ensuite QGIS qui est un logiciel de SIG Open Source que nous allons utiliser tout au long de notre étude. QGIS dispose de moins de 15 règles topologiques mais elles restent suffisantes pour des applications courantes et peuvent être combinées pour aboutir à des chaines de traitement de corrections. Enfin PostGIS est plus axé sur l'aspect gestion de base de données, mais grâce à des requêtes SQL il permet d'effectuer des manipulations des géométries.

Le logiciel développé et présenté par les auteurs apparaît comme celui proposant le meilleur choix avec non pas moins de 64 règles topologiques proposées pour aider à la correction des différentes erreurs.

Logiciel	Structure Géométrique	Etat des règles topologiques	Règles de topologies personnalisées	Nombre de règles topologiques
Arc/Info[1]	ArcNode	Très basique	Non	Très peu
ArcGIS 10.2	Similaire à OGC-SFA	Avancé	Non	32
GRASS 7	OGC-SFA Arc-Node	Très basique	Non	Très peu
Kosmo 3	OGC-SFA	Moyen	Non	<20
gvSIG 2.2	OGC-SFA	Moyen	Non	<20
QGIS 2.14	OGC-SFA	Moyen	Non	<15
PostGIS 2.2	OGC-SFA SQL/MM	Basique	Non	Peu
Oracle Spatial 11 g	OGC-SFA SQL/MM	Basique	Non	Peu
Radius Topology[2]	SQL/MM Oracle	Basique	Non	Peu
Developed System[3]	OGC-SFA over Jaspa	Avancé	Oui	67

Tableau 3 Liste des logiciels ayant des aides à la correction topologique (d'après J.Martinez-Lario et al. 2017)

II.2 La théorie des graphes

II.2.1 Histoire des graphes

Commençons par un bref historique ainsi qu'une définition des graphes. L'origine des graphes remonterait à un article présenté par le mathématicien Leonhard Euler qui a souhaité résoudre le problème des « Sept ponts de Königsberg » (Costes B. 2016). Le principe est simple, il souhaitait savoir s'il était possible de faire le tour de la ville en ne passant qu'une seule fois par chacun des ponts traversant la ville. Pour cela il a représenté les différentes parties de la

[1] Le logiciel Arc/Info n'est plus maintenu

[2] Le logiciel Radius Topology n'est plus maintenu

[3] Developed System correspond au logiciel développé par les auteurs de l'article.

ville par des nœuds reliés par des arêtes représentant les ponts, et il prouva que c'était impossible. Un autre problème qui a tenté d'être résolu au moyen de graphes est le problème du « cavalier ». Le but de l'opération était de savoir s'il était possible de déplacer un cavalier sur un échiquier en ne passant qu'une seule et unique fois sur chaque case. Cette fois-ci ce fut un succès, la démarche était possible.

« Les graphes constituent certainement l'outil théorique le plus utilisé pour les modélisations et la recherche des propriétés d'ensembles structurés. Ils interviennent chaque fois que l'on veut représenter et étudier un ensemble de liaison (orientées ou non) entre les éléments d'un ensemble fini d'objets » (Beauquier et al. (1992)). Ainsi, le modèle de graphe permet de représenter un réseau par un ensemble de nœuds (ou sommets) et d'arêtes lorsque les nœuds sont liés géographiquement. Dans notre cas, lorsque deux parcelles seront contiguës, les nœuds les représentants seront reliés par une arête.

II.2.2 Définition

Un graphe $G = (V, E)$ est constitué :

- d'un ensemble $V = \{v_1, v_2, ..., v_n\}$ dont les éléments sont appelés *sommets* ou encore *nœuds,*
- d'un ensemble $E = \{e_1, e_2, ..., e_n\}$ dont les éléments sont appelés *arcs* ou encore *arêtes*.

Un arc $e \in E$ relie deux sommets x et y de V et on note généralement $a = (x,y)$ ou encore $a = xy$. (B. Costes (2016)).

Un graphe se compose d'un graphe principal et d'un graphe dual. En théorie des graphes, le graphe dual d'un graphe est défini à l'aide des composantes de son complémentaire, lesquelles sont reliées entre elles en traversant les arêtes du graphe de départ (Wikipédia, l'encyclopédie libre 2018)

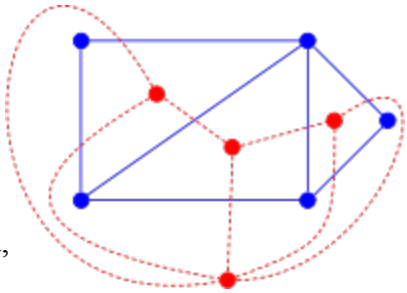


Figure 10 Graphe (en bleu) et graphe dual (en rouge)

II.2.3 Projets utilisant la théorie des graphes

Un projet proposé par l'Agence Nationale de Recherche et l'Université Toulouse 2, nommé ModelSpace, et Architerre son successeur, se sont donnés pour mission de retracer l'historique du parcellaire de certains départements au moyen de registres terriers et de

compoix[4]. Les terriers et compoix se présentent sous la forme de registres accompagnés parfois de plans permettant d'illustrer le parcellaire. Ces registres vont consigner le nom et la profession du propriétaire, parfois le nom de l'exploitant, mais surtout et très souvent la superficie, une localisation par un microtoponyme, les noms des propriétaires des parcelles adjacentes (Le Couédic et *al.* (2012)). Ainsi, en recoupant toutes les données fournies par les registres et en s'aidant des plans terriers lorsqu'ils sont disponibles ou à défaut du cadastre napoléonien, il est possible de cartographier le parcellaire de l'époque. Le modèle qui a été utilisé est donc le modèle des graphes d'adjacence décrit plus haut, les parcelles sont réduites à un point et deux parcelles voisines correspondent à deux sommets voisins dans le graphe.

Des questions en rapport avec les différents éléments appelés linéaires (cours d'eau, routes…), qui sont en réalité représentés par des polygones dans le plan cadastral, ont rapidement été soulevées. Plusieurs possibilités ont été formulées pour la prise en compte de ces linéaires :

- les linéaires seront représentés par un unique point, ce qui représentait le mieux la description fournie par les terriers et compoix ;
- enlever les linéaires et les laisser vides, le graphe sera alors déconnecté ;
- décomposer le linéaire en autant de segments qu'il y a de parcelles adjacentes ;
- décomposer le linéaire en autant de segments qu'il y a de parcelles adjacentes mais suivant sa longueur cette fois-ci ;
- supprimer les linéaires en les intégrant aux parcelles.

C'est cette dernière solution qui est retenue pour la construction des graphes dans le projet ModelSpace.

Nous avons pu également nous entretenir avec Kamaldeep Singh Oberoi doctorant dont le sujet de thèse repose sur la théorie des graphes. En effet, celle-ci traite de l'analyse des entités interagissant avec une route (piétons, véhicules, signalisation, trottoirs...). Grâce aux graphes, il est possible de savoir à un instant *t* où se situe n'importe quelle entité, dans quel sens elle se dirige et avec quelle autre entité elle possède des liens topologiques. Son travail introduit également la notion de granularité. Le fait d'avoir un nœud d'un graphe contenant plusieurs entités (une planche contenant des sections puis des feuilles et enfin les parcelles) se nomme

[4]« Les terriers correspondent à des sources fiscales privées tandis que les compoix sont des sources fiscales publiques présentes au sud de l'Europe » d'après Le Couédic et *al.* (2012).

graphe granulaire. Dans son cas, il réduit l'ensemble des piétons, des trottoirs et des véhicules d'un côté de la route à un nœud et il en fait de même pour l'autre côté. Ainsi il n'aura plus que deux entités à étudier.

Dans sa thèse « un modèle de graphe spatio-temporel pour représenter l'évolution d'entités géographiques », G. Del Mondo nous introduit notamment la relation entre le raisonnement spatial et temporel et la théorie des graphes pour nous mener aux graphes temporels. Elle se sert des graphes spatio-temporels pour modéliser l'évolution des entités à travers le temps. Cette théorie est intéressante à étudier puisque à la fin de notre étude nous disposerons de plusieurs graphes partageant une même zone géographique sur 3 époques différentes. Il nous faudra trouver un moyen de mettre en relation nos différents graphes afin de les comparer et de définir l'évolution qu'il y a pu avoir entre ces derniers. Dans son cas, elle montre également l'analyse de l'évolution d'un parcellaire en matérialisant les liens de filiations (division, fusion) à l'aide de graphes sur différentes périodes.

Le modèle des graphes sera donc intéressant à étudier lors de ce TFE, d'une part pour analyser la qualité du mosaïquage, d'autre part pour l'analyse de l'évolution du parcellaire.

II.3 Outils d'analyse

Dans cette partie seront décrits quelques outils utiles à la correction topologique. Nous ne développerons pas tout l'aspect technique des outils mais seulement leurs fonctions principales et les résultats qu'ils fournissent (Article : Description des outils d'analyse spatiale).

- <u>La triangulation de Delaunay</u>

La triangulation de Delaunay a été inventée par la mathématicien Boris Delaunay. La triangulation de Delaunay d'un ensemble P de points du plan est une triangulation DT(P) telle qu'aucun point de P n'est à l'intérieur du triangle circonscrit d'un des triangles de DT(P) (d'après la définition donnée par le site Wikipédia). La triangulation de Delaunay peut également être contrainte à l'aide de ligne de ruptures et ainsi suivre des formes imposées par l'opérateur.

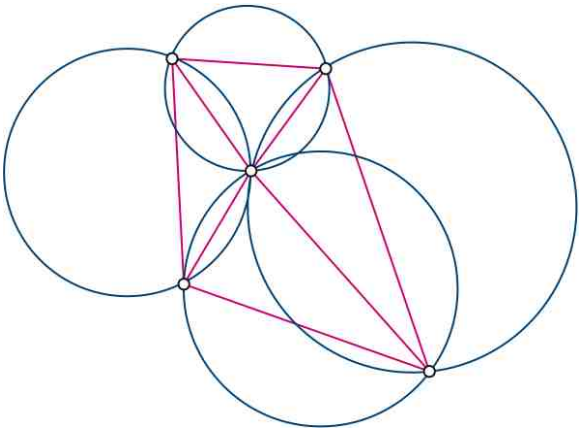


Figure 11 Triangulation de Delaunay, d'après Hervé Lehning

- L'enveloppe convexe

La triangulation de Delaunay, si elle n'est pas contrainte, est à la base de la création de l'enveloppe convexe. L'enveloppe convexe est générée à l'aide des points les plus extérieur de l'ensemble de points. Nous pouvons faire l'analogie avec un élastique que nous viendrions placer autour de l'ensemble de points. Ainsi, la forme donnée par l'élastique formera l'enveloppe convexe.

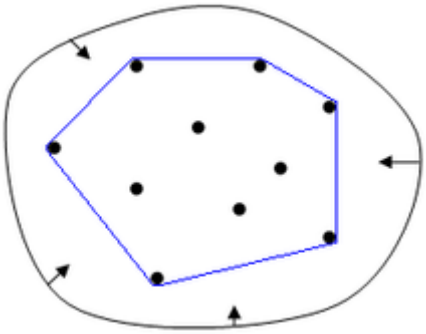


Figure 12 Analogie avec un élastique. La ligne bleue formant l'enveloppe convexe, d'après Wikipédia l'encyclopédie libre

- Union

L'outil d'union de QGIS permet de fusionner deux couches de forme en une seule. La nouvelle table attributaire créée après union, contiendra les champs des deux couches mères. Là où les planches s'intersectent il y aura la création d'un nouveau polygone qui prendra dans sa table attributaire les attributs des deux couches mères.

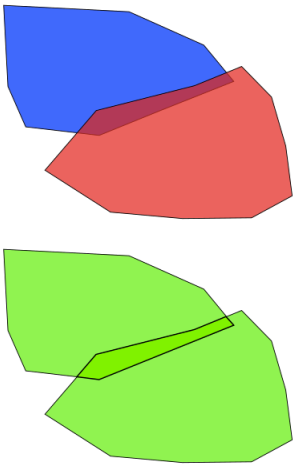


Figure 13 Illustration de l'union

- Intersection

L'outil d'intersection de QGIS permet d'extraire la ou les forme(s) créé(es) par la superposition de plusieurs couche. La nouvelle forme prendra les attributs des planches mères tout comme pour l'outil d'union.

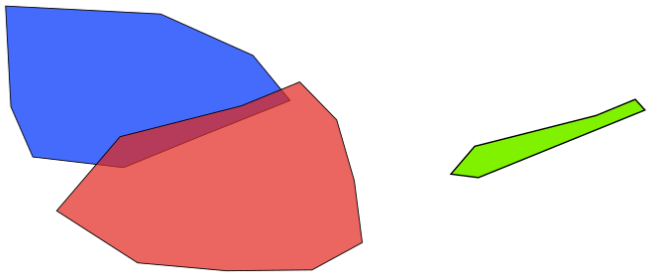


Figure 14 Illustration de l'intersection

Nous pouvons voir dans la figure 15 un exemple d'union de table attributaires. En haut à gauche la table attributaire de la couche 1, haut à droite la table attributaire de la couche 2 et enfin en bas, le résultat de l'union des tables attributaires. Les champs prendront une nouvelle nomenclature avec l'indice de la couche d'origine du champ.

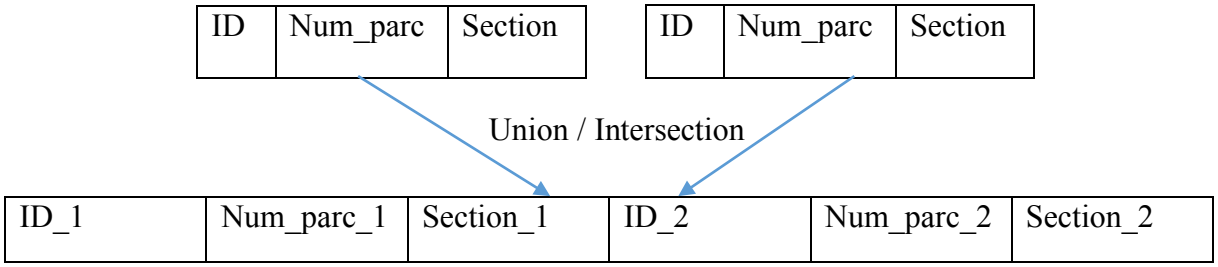


Figure 15 Exemple de fusion de tables attributaires

III. Méthodologie

III.1 Notions de relations entre les différentes entités

Dans la mesure où nous allons raisonner sur un problème concernant des relations topologiques, il est bon de s'intéresser aux relations pouvant lier les différentes entités que nous pouvons rencontrer sur les différentes planches. Les relations seront établies suivant le formalisme défini précédemment par le RCC8 de Randell et *al.*. Par la suite nous définirons les différents types d'entités, en relation avec notre travail, comme suit :

- Parcelle (P) (Polygone)
- Route (R) (Polygone)
- Cours d'eau (CdE) (Polygone)
- Bâtiment (B) (Polygone)
- Ecriture (E) (Polyligne)
- Limite de parcelle (L) (Polyligne)

X \ Y	Parcelle	Route	Cours d'eau	Bâtiment	Ecriture	Limite de parcelle
Parcelle	DC, EC, TPP, TPPi, NTPP, NTPPi	EC, DC	EC, DC	TPPi, NTPPi, EQ	NTPPi, TPPi, PO, NTPP, EC	EC, DC
Route	EC, DC	PO,EC, DC	PO,DC	EC, DC	NTPPi, TPPi, DC, PO, EC	EC, DC
Cours d'eau	EC,DC	EC, PO, DC	PO, DC	EC, DC	NTPPi, TPPi, DC, EC, PO	EC, DC
Bâtiment	TPP, NTPP, DC, EQ	EC, DC	EC, DC	DC, EC	DC, EC, NTPPi, TPPi, PO	EC, DC
Ecriture	TPP, DC, PO, NTPP, EC	TPP, DC, PO, NTPP, EC	TPP, DC, EC, PO, NTPP	DC, EC, PO, TPP, NTPP	DC, PO	EC, DC
Limite de parcelle	EC, DC	EC, DC	EC, DC	EC, DC	EC, DC, PO	PO

Tableau 4 Relations topologiques possibles entre différentes entités présentes sur une planche de cadastre

Afin de mieux corriger les erreurs, nous les avons tout d'abord identifiées. Il existe 6 types d'erreurs (non exhaustives) illustrées dans la figure 16 :

La première est rencontrée lorsque deux parcelles devraient se toucher mais qu'il y a un trou (lacune) entre elles (Cas n°1). La seconde est due au chevauchement de parcelles alors qu'elles devraient partager une frontière (Cas n°2). La troisième se produit lorsque qu'il y a deux parcelles jouxtant une route, qu'elles devraient partager une limite commune, mais qu'il y a un trou ouvert sur la route entre elles (Cas n°3). La quatrième qui s'apparente à la troisième, apparaît lorsque 2 parcelles devraient partager une frontière mais qu'elles sont séparées par un espace, faisant apparaître dans le graphe une adjacence indirecte alors qu'elle devrait être directe (Cas n°4). La cinquième que nous n'avons pas rencontrée correspond à un chevauchement de parcelles alors qu'il y a un espace non cadastré (route, rivière...) entre elles (Cas n°5). Enfin, la sixième est le cas extrême du cas n°5, la route est totalement recouverte par les parcelles qui se chevauchent (Cas n°6).

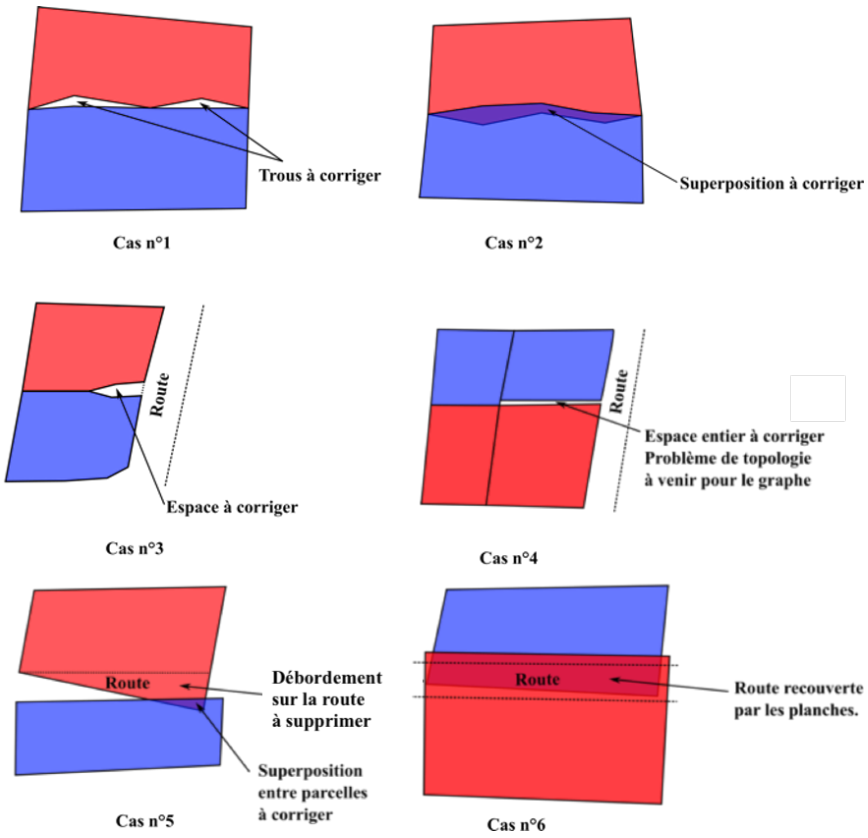


Figure 16 Différents cas de corrections topologiques à apporter

Après avoir vectorisé nos planches de cadastre, il a fallu les géoréférencer avec le programme de géoréférencement par transformation ridge par noyau gaussien. Seulement, ce procédé n'est pas suffisant pour mosaïquer. Les problèmes rencontrés après géoréférencement ici s'apparentent à des problèmes de relations spatiales entre les différentes entités, il s'agit de problème topologiques (superpositions, chevauchements...), ou de distance (espace entre deux planches alors qu'elles devraient se toucher). Pour corriger ces erreurs de superposition et de trous entre les planches, nous sommes en mesure de proposer une chaine de traitement à utiliser avec le logiciel QGIS. Cette chaîne de traitement est à utiliser plus particulièrement lorsque les planches sont totalement corrigées des erreurs de vectorisation (parcelles manquantes...).

Nous pouvons constater l'apparition des problèmes, sur la figure 17, entre deux planches différentes (ici en vert et bleu). Nous allons avoir le cas où les sections se chevauchent et le cas où elles sont séparées par un trou. Ces problèmes peuvent être corrigés à l'aide d'outils de correction topologiques Nous verrons par la suite comment utiliser les outils de QGIS pour régler les erreurs topologiques.

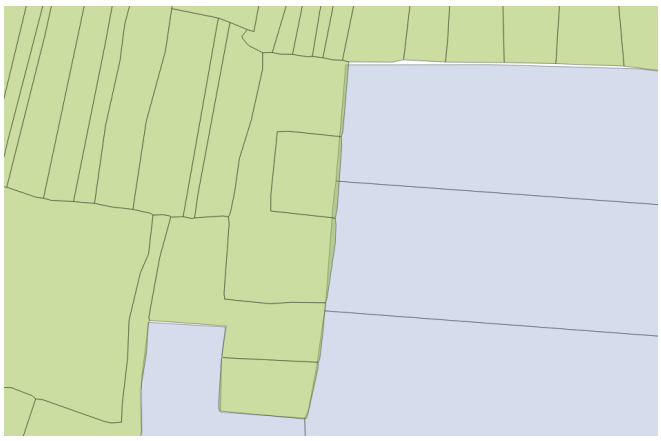


Figure 17 Problème de liaison entre parcelles

L'autre problème reste celui des espaces non cadastrés tels que les routes et rivières qui ne sont pas modélisées si ce n'est par des trous entre sections et parcelles. Les différentes options proposées plus haut restent à étudier.

III.2 Mosaïquage

Dans cette partie nous allons nous atteler à définir une chaine de traitement semi-automatisée sous QGIS afin de proposer un mosaïquage de nos planches vectorisées.

III.2.1 Présentation du traitement

Nous allons séparer le traitement du mosaïquage en deux étapes principales. La première consistera à supprimer les chevauchements des parcelles entre les différentes planches, et la seconde consistera à supprimer les trous. Nous avons tenté de proposer une chaine de traitement qui se veut la plus automatique possible, mais il reste tout de même des étapes manuelles à insérer dans la routine automatique comme par exemple le nettoyage du résultat de la triangulation de Delaunay pour former l'enveloppe de nos données. De plus des étapes pourraient être modifiées, notamment la partie suppression des chevauchements où nous pourrions nous passer de l'intersection et de la requête spatiale par une recherche de polygones possédant plus d'un attribut renseigné (voir partie II.3, Outils d'analyse). La figure ci-après illustre la chaine d'outils que nous avons mise en place pour effectuer le mosaïquage de nos données géoréférencées.

Diagramme de traitement

III.2.2 Suppression des chevauchements

Ici nous allons nous occuper de corriger les problèmes de chevauchements entres les différentes sections.

Tout d'abord commençons par quelques définitions des couches qui seront utilisées et créées :

- Planches : il s'agit des feuilles (ex : K2, J3...) définies dans la partie 1.2.2
- L'enveloppe : il s'agit ici d'une triangulation de Delaunay ayant les mêmes caractéristiques qu'une enveloppe convexe, mais dont certains triangles ont été supprimés pour coller au mieux à la forme des planches (voir définition triangulation de Delaunay partie II.3)
- Trous et réseaux : les trous correspondent aux espaces vides entre les planches là où elles devraient normalement se toucher. Les réseaux correspondent aux espaces non cadastrés tels que les routes et rivières. Ils seront matérialisés tout deux par des polygones (pour plus de détails voir partie III.1).

La figure ci-contre montre les planches géoréférencées sans aucune correction topologique. Nous pouvons donc apercevoir les trous et recouvrement entre les deux planches ici en bleu et rose.

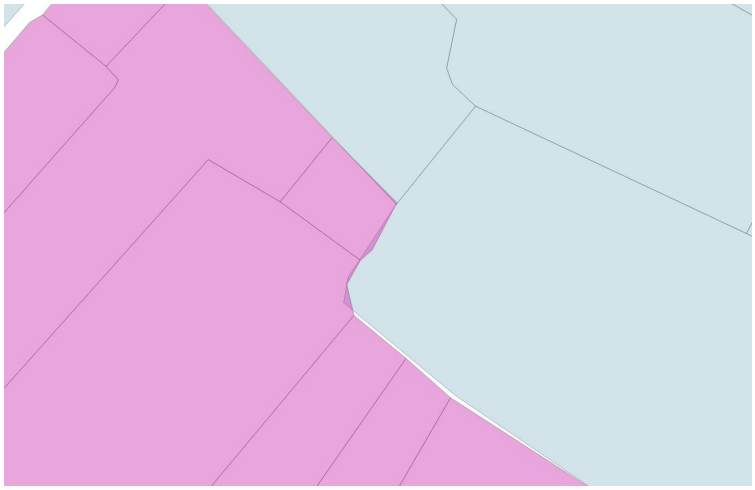


Figure 18 Planches géoréférencées non corrigées

Etape 1 : Vérifier les géométries

Cette étape est primordiale pour éviter des erreurs avec les outils QGIS pour les étapes suivantes. Elle se réalise avec l'outil *Vecteur/Outils de géométrie/Vérifier la validité*. Après cette étape il faut corriger les géométries invalides si nécessaire. La plupart des erreurs relevées par cet outil sont dues à un mauvais nettoyage de la vectorisation automatique (sommets en double...) et peuvent être corrigées automatiquement à l'aide d'outils QGIS 3.0.

Etape 2 : Intersection des planches

Ici nous allons récupérer les intersections des planches entre elles en créant une nouvelle couche de polygones issus de cette intersection. Cette couche nous servira par la suite pour effectuer une requête spatiale et détecter les chevauchements. Pour effectuer cette opération il faut utiliser l'outil *Vecteur/Outils de géotraitement/Intersection*. L'opération est à répéter autant de fois qu'il y a de planches qui s'intersectent. Par la suite nous allons réunir toutes les intersections dans une même couche avec l'outils *Vecteur/Outils de géotraitement/Union*. Il est possible que certaines géométries soient des multi-polygones alors qu'ils devraient être uniques, il faut donc utiliser l'outil *Vecteur/Outils de géométrie/De morceaux multiples à morceaux uniques* pour avoir un seul polygone par géométrie et ne pas avoir de multi-polygones. La sortie de cet outil est susceptible de contenir des doublons, c'est pourquoi il faut utiliser l'outil *Supprimer les géométries dupliquées*.

Etape 3 : Recherche des superpositions

Par la suite, il faut unir toutes les planches de la même année entre elles avec l'outil *Union* utilisé précédemment. La couche en sortie est de nouveau susceptible de contenir des géométries en double, il faut donc les supprimer avec l'outil *Supprimer les géométries dupliquées.* Ensuite, il faut effectuer une requête spatiale où l'union des planches contient, est égal ou est à l'intérieur des superpositions. Nous choisissons tous ces critères pour ne pas oublier de polygones, en effet normalement, le critère d'égalité devrait suffire à lui seul, mais à cause des différentes manipulations effectuées en amont, il y a des micro-erreurs qui se sont glissées dans nos polygones. C'est pourquoi nous choisissons ces 3 critères.

Etape 4 : Suppression des trous

Une fois les superpositions sélectionnées grâce à la requête spatiale, nous pouvons les fusionner avec un polygone voisin (plusieurs options sont proposées pour la fusion : avec le polygone le plus grand, avec le polygone le plus petit, avec le polygone ayant la plus grande frontière en commun) avec l'outil *Vecteur/Outils de géotraitement/Supprimer les débordements (2.18)* ou *Supprimer les polygones sélectionnés (3.0)*

Ici nous retrouvons le résultat de la première partie de la chaîne de traitement. Les chevauchements ont été corrigés. A la sortie de cette partie, il est possible d'avoir des parcelles en double, il faudra alors les nettoyer manuellement (voir partie IV, analyse des résultats). Il ne reste plus que les trous entre les planches.

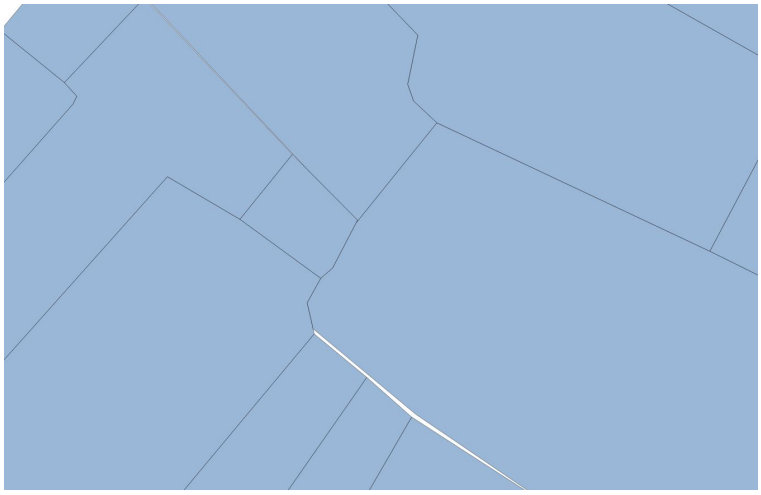


Figure 19 Planches corrigées des chevauchements

III.2.3 Suppression des trous

Dans cette partie, nous souhaitons corriger les problèmes de lacunes entre les parcelles et récupérer les espaces non cadastrés tels que les routes et cours d'eau pour les convertir en polygones.

Etape 1 : Création de l'enveloppe des planches

En premier lieu, nous avions pensé utiliser l'enveloppe convexe ou l'enveloppe concave pour réaliser cette étape, seulement aucune des deux ne retournait le résultat escompté. Nous avons donc décidé de nous baser sur la triangulation de Delaunay, sur laquelle s'appuie l'enveloppe convexe, qui pourra être adaptée pour coller aux limites des planches. Nous commençons par extraire tous les sommets des planches corrigées des chevauchements avec l'outil *Vecteur/Outils de géométrie/Extraction de nœuds (2.18)* ou *Extract vertices (3.0)*, cette étape est nécessaire pour la triangulation à venir. La réalisation de la triangulation de Delaunay avec les outils proposés par QGIS (*Vecteur/Outils de géométrie/Triangulation)* rencontre quelques problèmes, des triangles en recouvrent d'autres ou bien la triangulation est incomplète. C'est pourquoi il a été utilisé le logiciel Open Jump, puis par la suite un script Python, qui réalise une triangulation correcte avec l'outil *Outils/Générer/Triangulation*. Il faut bien penser à associer un système de projection (SRID : 2154) à la couche triangulée en passant par le menu *Couche/Définir le SRID...* La couche de triangulation est ensuite exportée au format shapefile pour continuer la chaine de traitement avec QGIS. Par la suite, une étape manuelle est nécessaire pour finaliser l'enveloppe, en effet, il faut passer la couche de triangulation en mode

d'édition et supprimer les triangles qui dépassent des planches tout en conservant les triangles qui forment les trous et réseaux.

Etape 2 : Récupération des trous et réseaux

Une fois les triangles superflus supprimés nous allons utiliser l'outil *Vecteur/Outils de géotraitement/Dissolve (2.18)* ou *Regrouper (3.0)* sur la couche de triangulation nettoyée ainsi que sur la couche des planches corrigées des chevauchements. Pour récupérer les trous et les réseaux, nous allons utiliser l'outil *Vecteur/Outils de géotraitement/Différence*, avec en entrée la couche des triangles regroupés et comme couche de découpe la couche des planches regroupées. Les trous et réseaux ainsi récupérés correspondent aux lacunes entres planches matérialisées par de nouveaux polygones ainsi que les routes et cours d'eau matérialisés eux aussi par des polygones.

Etape 3 : Suppression des trous et conservation des réseaux

Pour pouvoir supprimer les trous, nous allons utiliser une méthode similaire à celle que nous avons utilisé pour supprimer les chevauchements. Pour ce faire, nous allons unir la couche des planches corrigée des chevauchements (obtenue à l'étape 4 de la correction des chevauchement) avec la couche des trous et réseaux récupérée dans l'étape précédente. Nous pouvons ainsi appliquer une requête spatiale là où la planche corrigée des chevauchements unie à celle des trous et réseaux contient, est égale ou est à l'intérieure de celle des trous et réseaux. Nous appliquerons ensuite une recherche par expression pour sortir le réseau de la sélection. En effet, celui-ci ayant une superficie plus grande que celle des trous entre planches, nous pouvons trier la sélection de la requête spatiale avec l'expression *$area < 100* (ce seuil est choisi empiriquement, il est suffisamment grand pour ne pas prendre en compte les trous, et suffisamment petit pour exclure le réseau de la sélection).

Suite à ces requêtes, il ne restera donc plus que les trous entre planches de sélectionnés. Nous pouvons donc utiliser l'outils *Supprimer les débordements* précédemment utilisé pour corriger les trous. Nous obtenons ainsi une couche totalement mosaïquée.

Ici nous avons le résultat final. Les planches sont correctement mosaïquées entre elles, les trous et les chevauchements ont été correctement traités. Seul le cas n°3 présenté dans la partie 3.2 n'est pas corrigé, mais cela ne pose pas de problème pour la suite du processus. Certaines parcelles auront leurs géométries qui changent, toutefois n'affectera pas la génération de graphe puisque celle-ci prend seulement en compte la topologie des parcelles.

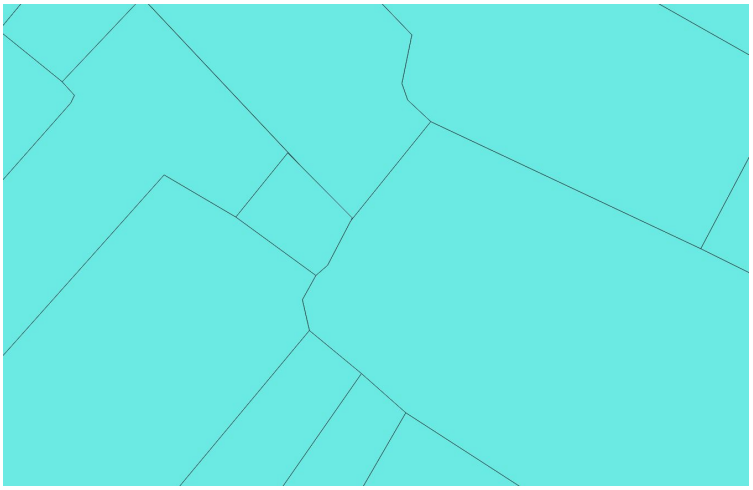


Figure 20 Planches corrigées des chevauchements et des trous

Etape 4 : Récupération du réseau comme couche à part entière

Enfin pour récupérer le réseau, il suffit d'effectuer une requête spatiale sur la couche mosaïquée où elle est égale à la couche des trous et réseaux. Puisque cette couche est corrigée des trous, la sélection effectuée ne retournera que le réseau. Il ne reste plus qu'à enregistrer les entités sélectionnées comme nouvelle couche).

III.3 Application de la théorie des graphes

III.3.1 Différents types de graphes

Afin d'étudier les relations de voisinage entre les différentes entités, nous pouvons proposer différents graphes.

- <u>Graphe planaire</u>

Le premier est le graphe formé naturellement par les limites de parcelles, où les nœuds sont les intersections entre limites de parcelles et les arrêtes sont les limites reliant ces sommets, il est appelé graphe planaire. Il ne servira pas dans l'étude des relations de voisinage. Son seul rôle sera de simplifier les géométries, de réduire les différents segments brisés à des segments rectilignes avec seulement deux nœuds : le départ et l'arrivée. Il ne servira pas dans notre étude mais il est intéressant de connaître son existence.

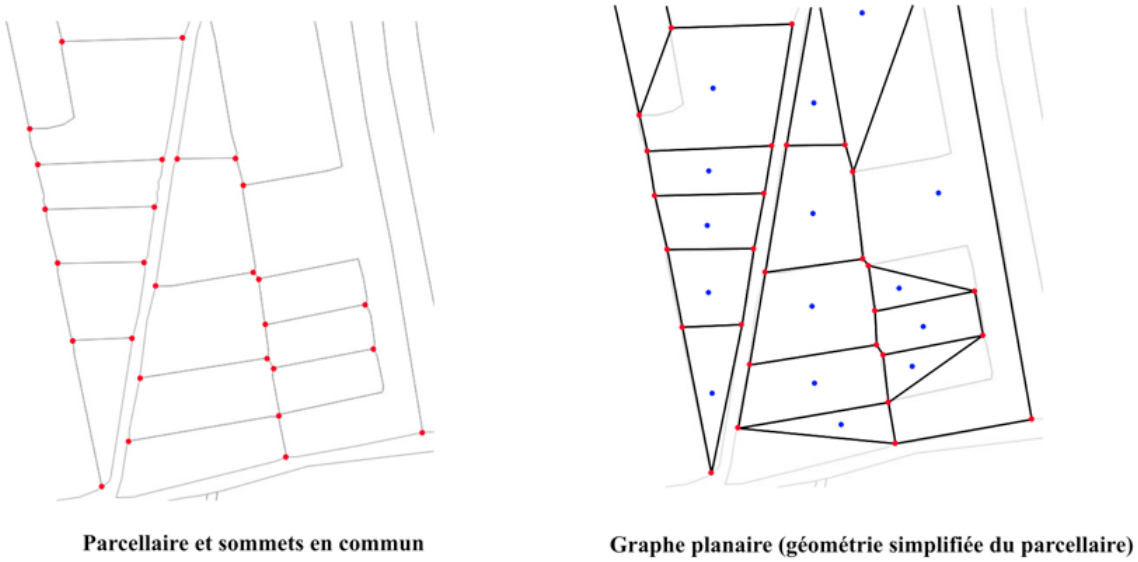


Figure 21 Graphe planaire du parcellaire

- <u>Graphe dual / graphe d'adjacence</u>

Le second est défini par les relations d'adjacences entre deux parcelles, modélisées par leurs centroïdes. Ici le nœud sera donc le centroïde de chaque parcelle et les arêtes correspondront aux relations de voisinage entre ces dernières. Ce graphe sera utile pour étudier les relations d'adjacences directes entre deux parcelles. Nous disposons d'une fonction implémentée sous Python pour calculer ce dernier à l'aide du calcul d'une matrice d'adjacence. Il constitue également le graphe dual du graphe planaire.

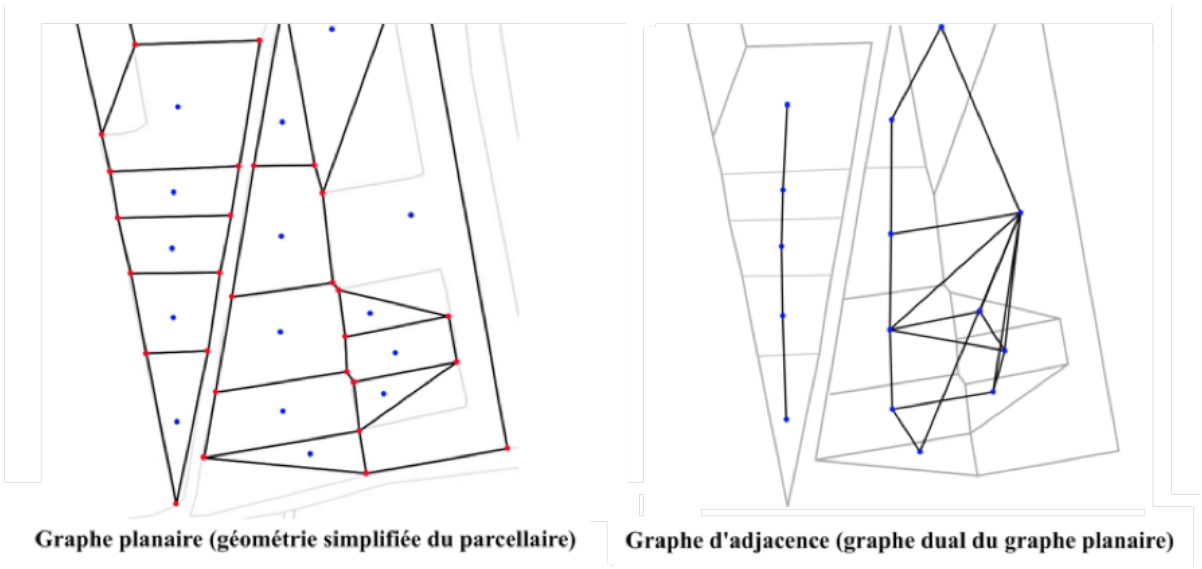


Figure 22 Graphe dual (adjacence) du graphe planaire

- Graphe temporel

Un des graphes les plus intéressants à étudier est le graphe temporel. Il permettra d'étudier l'évolution des parcelles, en modélisant les changements d'états (division, fusion...).

Comme nous pouvons le voir sur l'illustration ci-contre, grâce aux graphes temporels, il est facile de constater qu'entre l'année X et l'année Y, la parcelle A s'est divisée en parcelles C et D et que la parcelle B n'a pas subi de changement d'état. Puis entre les années Y et Z, la parcelle C n'a pas changé d'état, que la parcelle B a été divisée en deux parties : l'une formant une nouvelle parcelle F et l'autre fusionnant avec la parcelle D pour former E.

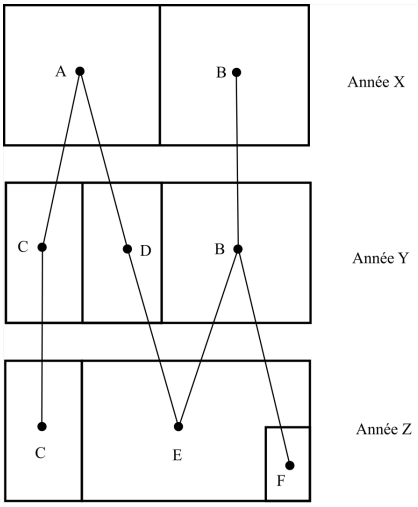


Figure 23 Exemple de graphe temporel, d'après G. Del Mondo (2011)

Le graphe temporel pourra s'avérer très utile dans la partie suivante de notre étude, celle de la création de la base de données dédiée au suivi de l'évolution du parcellaire.

- Graphe granulaire

Nous avons également une relation de granularité entre les différents niveaux d'« échelle » du cadastre. Nous avons trois niveaux différents suivant les différentes échelles vues précédemment dans la figure 3 :

- Le tableau d'assemblage dans lequel sont regroupées les sections ;

- Les sections, qui sont composées de feuilles ;

- Les feuilles, qui sont composées de parcelles et des espaces non cadastrés ;

- Les parcelles qui contiennent ou non des bâtiments, et autres informations (n° de parcelle, lieux dits...).

Ainsi, nous pourrions également générer des graphes d'adjacence dits granulaires au niveau du tableau d'assemblage où chaque nœud contiendra le graphe des feuilles et ainsi de suite jusqu'à arriver au niveau des graphes d'adjacence des parcelles.

Il n'en reste pas moins que tant que nous aurons des erreurs de topologie (trous entre planches et parcelles, superposition de parcelles…), nous ne serons pas en mesure de créer un graphe d'un seul tenant en une seule fois. Ou alors celui-ci serait entaché d'erreurs dues à la présence de polygones superflus tels que les superpositions ou les trous entre parcelles. La solution serait dans un premier temps de créer les graphes des différentes planches à notre disposition, puis de les mosaïquer.

Une autre solution serait de modéliser les espaces non cadastrés entre les différentes entités à l'aide de la création d'enveloppes convexes, ou d'une triangulation de Delaunay, et de la superposer avec les différentes planches. Ainsi, nous obtiendrions un ensemble de parcelles sans trou. Il restera le problème des chevauchements. Sous QGIS, le problème est complexe, il est possible de fusionner différentes planches, seulement leurs chevauchements entraineront la création d'une nouvelle entité. Il est par la suite possible de supprimer ces entités par une requête spatiale sur le critère d'une surface minimale et de fusionner cette surface avec un polygone adjacent (le plus grand, le plus petit, ou celui ayant la plus longue limite commune). Seulement, cela pourrait entrainer la disparition de petites parcelles prises en compte lors de la requête spatiale basée sur la surface.

Une fois le graphe spatial de chaque époque effectué, nous serons en mesure de réaliser une comparaison entre les différents graphes et d'ainsi générer un graphe temporel où les nœuds sont les parcelles à un instant *t* et les branches correspondront à un changement d'état (ou non) de la parcelle.

III.3.2 Traitement d'une planche

Ici nous allons nous pencher sur le cas où les graphes seront générés planche par planche. Ici on ne parlera que d'adjacences directes entre parcelles. Le but de cette approche est d'avoir les graphes d'adjacence de chaque planche géoréférencée et de les assembler. Nous obtiendrons ainsi un ensemble de graphes déconnectés. Au sein de chaque planche nous aurons également plusieurs graphes déconnectés dus au fait que les routes et cours d'eau vont séparer les planches en différents îlots de parcelles.

Notre première idée était de créer une matrice d'adjacence codée en binaire avec 1 si les polygones sont voisins et 0 s'ils ne le sont pas. Par la suite, nous avons trouvé une bibliothèque d'analyse spatiale nommée « Pysal » avec un module « queen_from_shp » qui permet de créer un graphe d'adjacence de nos parcelles. Le module demande en entrée un fichier de forme de type polygones.

Polygones	1	2	3
1	0	1	0
2	1	0	1
3	0	1	0

Tableau 5 Exemple de matrice d'adjacence

Nous avons pu relever différents problèmes en générant les premiers graphes.

Le premier est lié à des artefacts restants après vectorisation correspondant à des surfaces dont l'aire est inférieure à 1m². Il va générer des centroïdes excentrés. En effet à cause des petites surfaces, la génération des centroïdes ne va pas se faire correctement sous python. Pour corriger ce problème, il faut utiliser un algorithme qui va détecter les polygones concernés et les fusionner avec les polygones voisins (v.clean de GRASS avec l'outil rmarea). Nous remarquons qu'en utilisant les centroïdes générés par QGIS, le problème ne se présente pas, il serait donc associé à un souci numérique de Python lors du calcul de centroïdes avec de petites aires.

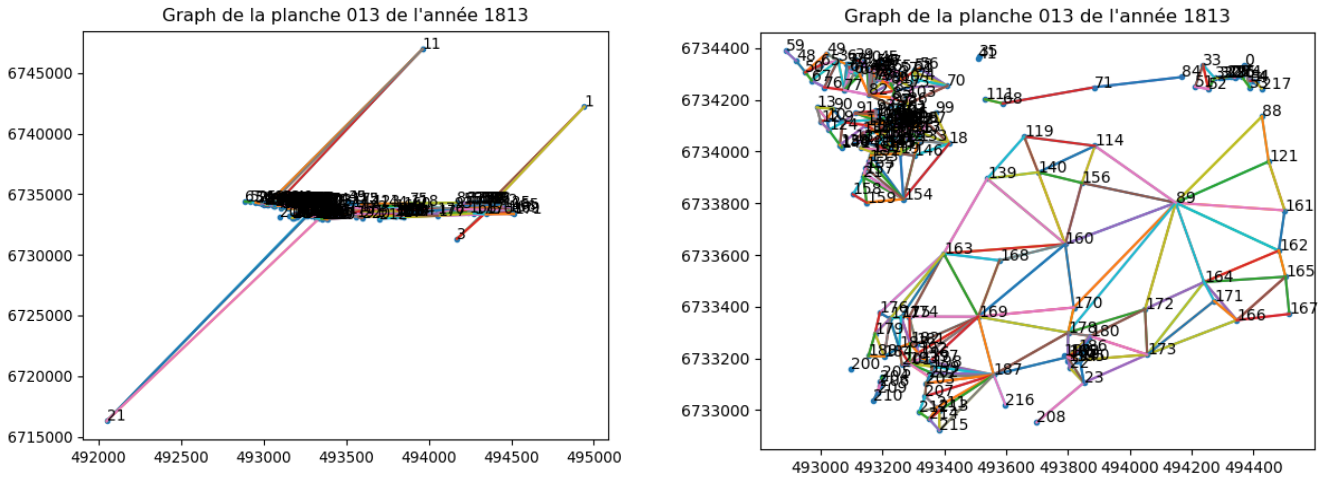


Figure 24 Graphe d'adjacence de la planche 013 avant/après correction

Certaines planches ne peuvent pas être lues par la fonction « queen_from_shp ». En effet, Python nous retourne l'erreur « struct.error: unpack requires a buffer of 44 bytes ». Nous pensons qu'il s'agit d'un problème de lecture du format binaire shapefile. Le fait est que si l'on applique la méthode de nettoyage des petites aires inférieures à 1m² utilisée précédemment pour régler le problème des centroïdes excentrés, le problème se résout de lui-même et on peut générer le graphe de la planche. Pour générer le graphe correctement, il faut donc procéder à un nettoyage topologique minutieux des planches

III.3.3 Graphe de plusieurs planches

Afin de lier les différents graphes entre eux, nous avons choisi d'affecter à notre matrice d'adjacence avec différents poids suivant les différentes relations spatiales qui unissent deux parcelles :

- 0 si les parcelles ne sont pas adjacentes, c'est à dire qu'elles ne partagent pas de limite commune (intersection vide);
- 1 si la distance la plus courte entre deux limites de parcelles est nulle (les parcelles partagent une limite séparative) ;
- 2 si les parcelles se chevauchent (intersection non vide) ;
- 3 si la distance la plus courte entre deux limites de parcelles est inférieure à un certain seuil et supérieure à 0 (cas d'une route ou d'une rivière passant entre les parcelles).

Il y aurait donc des tests de plus courte distance entre polygones à réaliser. Grâce à cette matrice d'adjacence, nous serions en mesure de définir un graphe contenant les différents graphes de chaque planche vectorisée. De plus il serait possible de donner des informations sur les arêtes du graphe, en caractérisant les liens qui les unissent pour indiquer si les parcelles sont connectées ou séparées par une route ou un cours d'eau.

Une autre approche que nous sommes en mesure de proposer se manifeste après correction des erreurs topologiques. Une fois ces défauts corrigés et les réseaux obtenus (voir méthodologie dans la partie III.3), nous pourrons plus facilement effectuer un graphe d'adjacence. Pour ce faire, nous triangulons le réseau avec la méthode de la triangulation de Delaunay et si un sommet de triangle est en commun avec un sommet d'une parcelle et qu'un autre sommet de ce même triangle est en commun avec le sommet d'une autre parcelle, nous pourrons relier ces dernières dans notre graphe avec comme relation d'adjacence « voisines au travers d'une route/rivière ».

C’est cette dernière méthode que nous retiendrons. Afin de connaître les différentes adjacences directes (parcelles contigües) et indirectes (route ou cours d’eau entre les parcelles), nous avons donc basé notre réflexion sur la triangulation de Delaunay. Une fois les planches corrigées de tous les défauts (chevauchements et lacunes) et les routes et cours d’eau récupérés (méthode détaillée dans la partie 4 Mosaïquage), nous triangulons ces derniers à l’aide de la triangulation de Delaunay. Par la suite, grâce à un script Python, nous générons une nouvelle matrice d’adjacence qui prendra la valeur 2 sur le critère : « deux parcelles non connexes ont une relation de connexion (partagent un sommet) avec le même triangle de route ou de rivière ». Ensuite, nous pouvons récupérer la matrice issue de la fonction *queen_from_shapefile* de Pysal qui est composée de 1 et de 0 et la compléter avec notre nouvelle matrice d’adjacence.

Seulement, un problème est rapidement apparu. En effet, il peut arriver que d’un côté de la route nous ayons de nombreuses parcelles et de l’autre côté une unique parcelle ne partageant que 2 sommets avec la route car elle possède une géométrie très rectiligne sur une de ses faces. A ce moment-là, les triangles n’auront pas de sommets en communs avec la grande parcelle et les parcelles situées de l’autre côté de la route comme en témoigne l’illustration ci-dessous. La seule solution est de densifier le nombre de sommets en en rajoutant un ou deux entre deux sommets consécutifs des parcelles et du réseau. Le programme Python est lui établi de telle sorte à ce que « si un sommet d’une parcelle touche un triangle et qu’un autre sommet d’une autre parcelle touche également ce triangle alors celles-ci seront adjacentes indirectement ». De plus le manque de sommets va impliquer des adjacences entre deux parcelles consécutives qui ne devraient pas avoir lieu. En effet, la triangulation de Delaunay ici effectuée se base sur les sommets existants, mais va également interpoler des points sur les segments pour coller au mieux à la forme proposée, elle ne va cependant pas créer de nouveau sommet. C’est pourquoi il est important d’augmenter la densité des points.

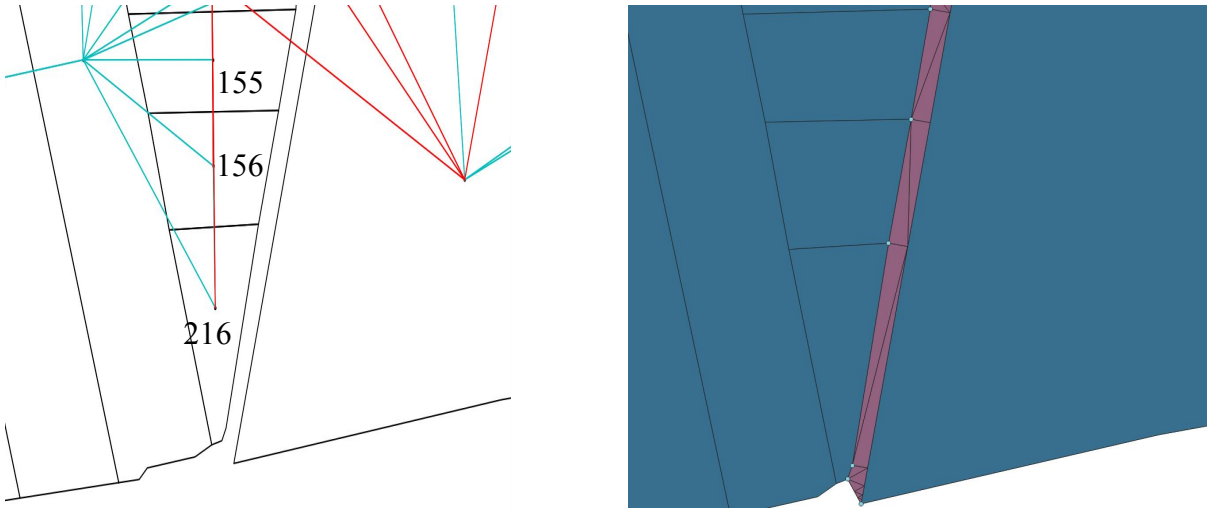


Figure 25 Problème de densification des sommets

Ainsi, en densifiant le nombre de sommets, (ici on rajoute deux sommets entre deux sommets consécutifs), nous allons pouvoir rajouter les adjacences manquantes et supprimer les aberrations telles que la parcelle 216 et 155 qui sont voisines alors qu'il y a la parcelle 156 entre elles.

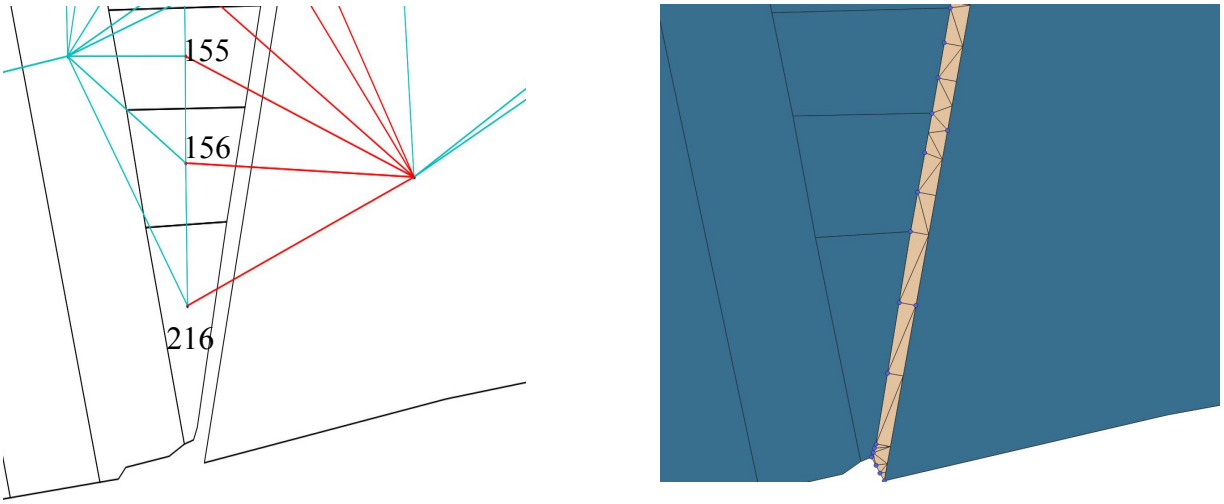


Figure 26 Résolution du problème d'adjacences indirectes

- <u>Méthodologie utilisée dans le script Python de génération de graphes</u>

Le programme est conçu pour afficher les adjacences directes et indirectes, c'est à dire au travers d'un espace non cadastré. Les bibliothèques utilisées sont les suivantes :

- Shapefile, qui est une bibliothèque permettant la manipulation de fichiers .shp sous python (elle nous sert pour l'écriture de fichiers shapefiles);

- Os, qui permet de manipuler les répertoires de fichier ;
- Pysal, qui est une bibliothèque d'analyse spatiale (pour utiliser le module de génération de graphe d'adjacence);
- Fiona, qui permet également la gestion des fichier shp sous python (elle nous sert pour l'ouverture et la lecture de fichiers shapefiles.

La première étape est d'importer nos fichiers de format shapefile (parcellaire et réseau triangulé) à l'aide du module *Reader* de la bibliothèque shapefile. Par la suite on va créer une liste contenant autant de listes vides qu'il y a de parcelles. Cette liste nous servira à stocker les indices des triangles qui ont un sommet en commun avec une parcelle, pour chaque parcelle. Ensuite, nous remplissons cette liste en testant tous les sommets des parcelles et ceux des triangles. Si les coordonnées concordent, on ajoute à la liste de la parcelle, l'indice du triangle concerné.

Par la suite, on crée une matrice nulle de dimension NxN (avec N le nombre de parcelles). Nous testons toutes les listes pour voir quelles parcelles ont des triangles en commun. Si 2 parcelles n et m partagent le même indice de triangle, la matrice à l'indice [n, m] prendra la valeur 2. Il faudra par la suite mettre toutes les valeurs de la diagonale de cette matrice à 0 puisque la diagonale correspond à la relation d'adjacence d'une parcelle avec elle-même. Or une parcelle ne peut pas être voisine avec elle-même.

Après cela, nous utilisons le module *queen_from_shapefile* de la bibliothèque Pysal pour calculer les relations d'adjacences directes. Nous récupérons la matrice d'adjacence issue de ce calcul, qui est de même dimension que la matrice d'adjacence indirecte calculée plus haut, et nous la complétons de la manière suivante :

- Si la valeur de la matrice d'adjacence directe à l'indice [m, n] est égale à 0 et que la valeur de la matrice d'adjacence indirecte est 2 à l'indice [m, n], alors la matrice d'adjacence directe prendra la valeur 2 à l'indice [m,n] ;
- Si la valeur de la matrice d'adjacence directe à l'indice [m,n] est égale à 1 et que la valeur de la matrice d'adjacence indirecte est 2 à l'indice [m,n], alors la matrice d'adjacence directe gardera la valeur 1 à l'indice [m,n]. Le but est d'ici de conserver les adjacences directes qui ont été calculées comme indirectes car deux parcelles consécutives jouxtant une route ou un cours d'eau partagent forcément un sommet de triangle.

Nous avons donc notre matrice d'adjacence comprenant les adjacences directes et les adjacences indirectes, avec pour valeur 1 si des parcelles sont directement voisines, et 2 si elles ne sont pas directement voisines.

Il ne nous reste plus qu'à afficher les relations d'adjacences sous la forme d'un graphe sur une figure. Pour se faire nous utilisons la bibliothèque matplotlib. Nous afficherons les centroïdes des parcelles avec leurs numéros, les limites de parcelles et enfin les relations d'adjacences, en bleu pour les directes et en rouge pour les indirectes.

IV. Présentation des résultats[5]

IV.1 Les résultats du mosaïquage

IV.1.1 Année 1813

Concernant l'année 1813, nous n'avons pas réussi à fournir de résultats concernant le mosaïquage. Nous avons tout de même effectué la vectorisation des cinq planches ainsi que le géoréférencement de celles-ci tout comme pour les années suivantes. Nous pouvons tout de même analyser le fait que le géoréférencement provoquait de fait des soucis de relations topologiques importants, comme nous pouvons le constater par transparence sur la figure 27 ci-dessous. Cela est dû au fait que le parcellaire à énormément changé entre la donnée de 1813 à géoréférencer et la donnée du PCI-vecteur de 2016 de référence permettant le géoréférencement. En effet, la modification du parcellaire par des remembrements et des aménagements urbains (routes, lotissements...) à diminué le nombre de points d'amers entre les deux jeux de données. Ainsi, la qualité du géoréférencement a été dégradée, comparée aux jeux de données de 1850 et de 1972-74. Le résultat global du géoréférencement des cinq planches de 1813 est consultable en annexe 1.

Figure 27 Extrait du géoréférencement des planches de 1813

[5] Les résultats sont analysés dans les parties 4.1, 4.2 et 4.3 avec des vues locales sur les éléments corrigés et les problèmes rencontrés. Les résultats globaux des différentes époques seront disponibles en annexes 1, 2 et 3 de ce document.

Nous n'avons pas pu faire fonctionner la chaîne de traitement QGIS sur ces planches est à cause de la présence d'erreurs géométriques que nous n'avons pas décelées et que n'avons pas pu corriger ce qui a provoqué un arrêt de la chaine dès le début de son utilisation. Si l'erreur est corrigée, la chaine de traitement pourra alors être lancée et le mosaïquage effectué.

IV.1.2 Année 1850

Ici, la chaine de traitement a fonctionné en partie. Les chevauchements ont tous été corrigés mais les trous n'ont été corrigés que partiellement, si bien que certains d'entre eux sont restés entre les planches sous la forme de polygones. Nous pensons qu'il s'agit d'un problème apparu lors de la sélection de ceux-ci avant leur fusion avec les parcelles voisines. Le problème est illustré ci-dessous dans la figure 28.

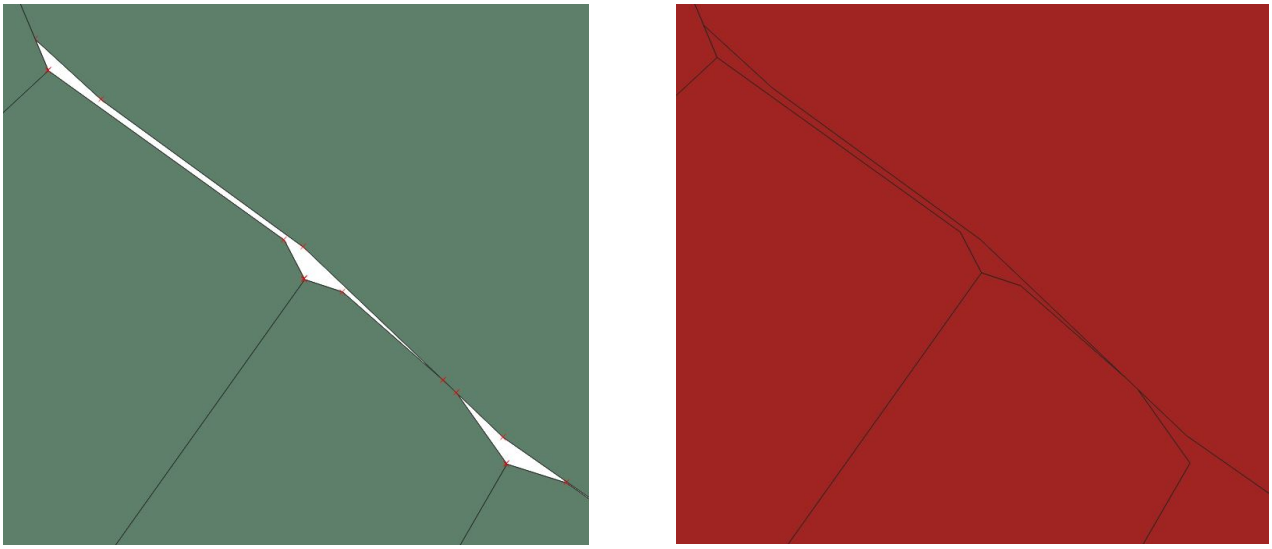


Figure 28 Planches corrigées des chauvauchement (à gauche) planches partiellements corrigées des trous (à droite)

Un autre souci est décelé à la sortie de la première phase de la chaîne de traitement (correction des chevauchements) en effet des parcelles à l'origine de chevauchements se retrouvent dupliquées partiellement ce qui donne l'impression que le chevauchement n'a pas été corrigé. Nous pouvons constater ce problème en augmentant la transparence de la couche corrigée des chevauchements. Si l'on supprime la parcelle dupliquée, le problème est réglé.

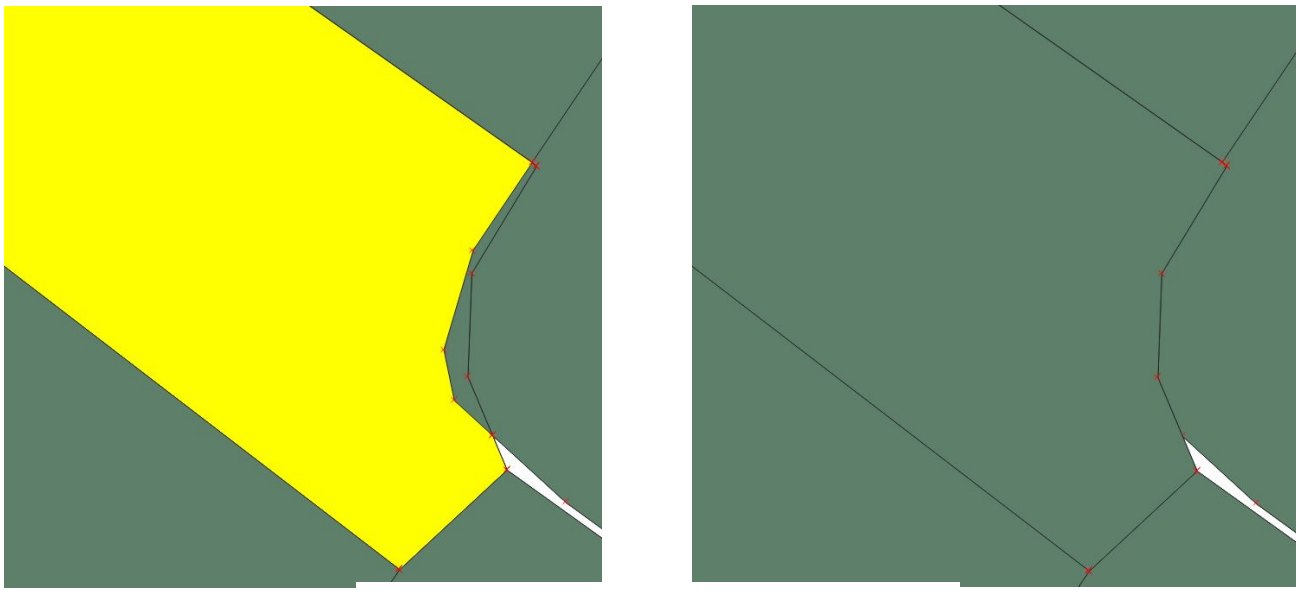


Figure 29 Problème de parcelle dupliquée (ici en jaune)

Si l'on met ces problèmes de côté, le mosaïquage pour les planches de cette année-là se fait de manière plutôt correcte. Cela montre encore une fois qu'il faut avoir des données, en amont du mosaïquage, qui soient correctement préparées (nettoyage géométrique, vectorisation correcte...).

IV.1.3 Années 1972-1974

Dans l'étude de ces années, nous avons des données plus récentes et mieux conservées, si bien que la vectorisation est quasi-parfaite pour lesquelles il y'a eu moins de nettoyage post-vectorisation à effectuer. Ainsi, le mosaïquage s'est effectué sans encombre à ceci près, que le problème des parcelles dupliquées en sortie de la première phase de la chaîne de traitement apparaît de nouveau. Encore une fois, dès que nous supprimons les parcelles incriminées, le mosaïquage se fait correctement.

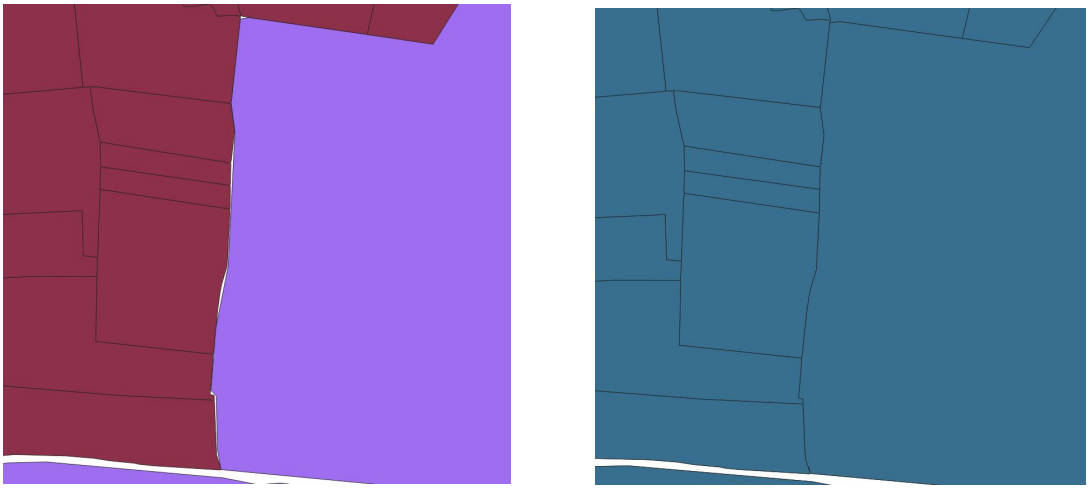


Figure 30 Mosaïquage 1972-74 avant/après

Nous pouvons ainsi conclure sur le fait que le mosaïquage va s'opérer plus ou moins correctement suivant l'année des données utilisées. Nous aurons ainsi de meilleurs résultats pour les planches de 1972-74 qui ont été parfaitement conservées.

La bonne marche de la chaîne de traitement est donc directement liée à la vétusté des planches de cadastre et à l'attention que nous porterons au nettoyage en post-traitement des données vectorisées. La chaîne d'outils proposée dans ce TFE reste encore à améliorer, notamment concernant le problème des parcelles dupliquées en sortie de la première phase, toutefois elle reste tout à fait utilisable et fonctionnelle.

IV.2 Les résultats des graphes

IV.2.1 Année 1813

Comme dit précédemment, nous n'avons pas pu générer le mosaïquage de cette époque-ci. Par conséquent, nous ne sommes pas en mesure de proposer le graphe complet du mosaïquage de cette année. Nous avons tout de même les graphes de chaque planche individuellement.

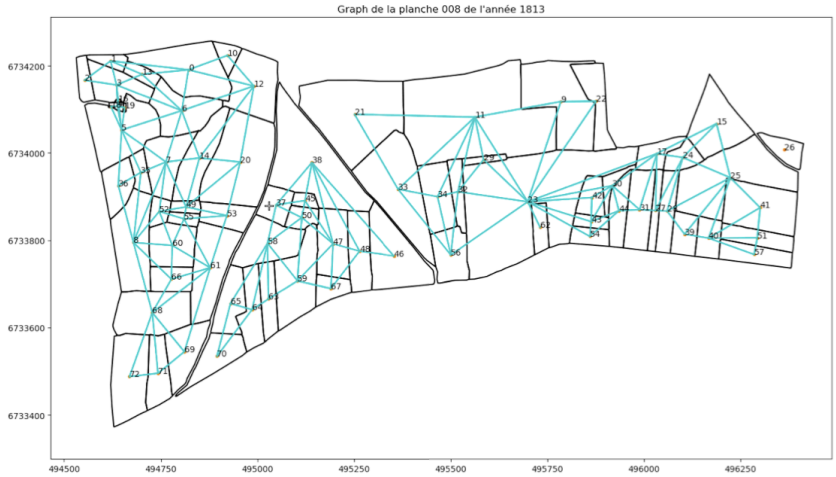


Figure 31 Exemple de résultat pour le graphe d'adjacence d'une planche de 1813

IV.2.2 Année 1850

Ici, le graphe a pu être généré mais il reste entaché par des erreurs décelées plus haut tels que les trous non corrigés où encore des manques d'adjacences indirectes dues à un manque de sommets sur les routes et rivières et les parcelles. Pour résoudre ce problème et s'affranchir de la densification de sommets, il faudrait ajouter une condition dans le script Python, telle que :

« si un triangle d'une route ou d'une rivière possède un sommet sur une limite d'une parcelle et que ce sommet n'est pas en commun avec le sommet de la parcelle, alors ce triangle est considéré comme un élément d'adjacence indirecte avec la parcelle d'en face ». Cette règle pourrait se baser sur des éléments de colinéarités des coordonnées de deux sommets successifs d'une parcelle et du sommet du triangle considéré, et le sommet de triangle devra se trouver dans la boite englobante formée par les sommets successifs de la parcelle.

IV.2.3 Année 1973

Ici, le seul souci rencontré concerne le manque de densification de sommets. La solution proposée dans la partie IV.2.2 reste également envisageable pour les années 1972-74. Nous pouvons constater un exemple d'une adjacence qui n'a pas été détectée par manque de densification des sommets sur la figure 31 en magenta.

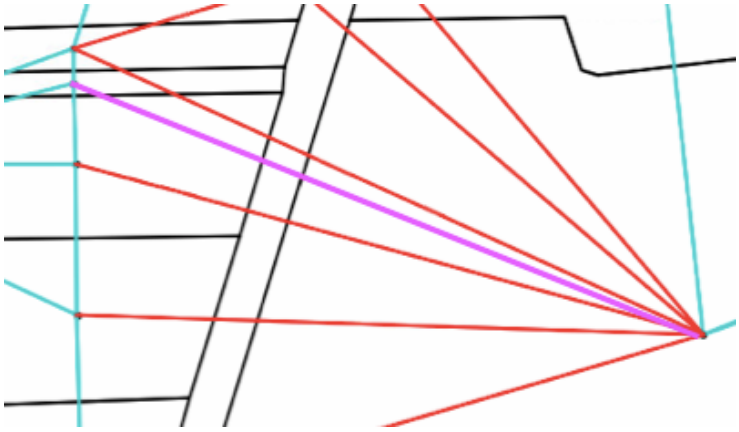


Figure 32 Exemple d'adjacence manquée dans le programme Python (ici en magenta)

IV.3 Liens entre mosaïquage et théorie des graphes

A la suite des travaux entrepris dans les parties précédentes, nous sommes donc en mesure de présenter différents graphes à différents niveaux (notion de granularité). Nous avons donc les graphes à l'échelle de la commune, où chaque nœud est une section, les graphes à l'échelle de sections où chaque nœud est une feuille et les graphes à l'échelle des feuilles où chaque nœud représente une parcelle.

Grâce au modèle présenté dans la partie III.3, nous pouvons réaliser le mosaïquage de nos sous-sections et en extraire un graphe d'adjacences (directes ou indirectes). Ce graphe peut

encore comporter des erreurs dues à des problèmes non décelés en amont ou que nous n'avons pas pu corriger, ou encore une triangulation de Delaunay du réseau imparfaite. La notion de granularité pourrait nous aider à déceler ces erreurs. En effet, si au niveau supérieur du graphe nous décelons qu'il est censé y avoir un écart entre deux sous-sections mais que celles-ci se touchent au niveau inférieur du graphe granulaire, le graphe pourrait alors nous aider à déceler une potentielle erreur. Cela suggère qu'en amont du traitement les espaces non cadastrés aient été répertoriés et correctement identifiés.

Il serait également intéressant de créer un système de règles à l'aide d'une pré-analyse des documents. En effet, nous pourrions dire que sur certaines bordures de parcelles il y a des routes ou des cours d'eau et qu'ainsi, il est impossible que les planches se superposent à cet endroit. Il en va de même pour les endroits où les planches devraient coïncider, si au moment du mosaïquage il apparaît qu'il y a un trou alors qu'il ne devrait pas y avoir d'espace alors il faudra appliquer une correction. Cet ensemble de règles permettrait d'éviter les erreurs dues à la correction automatique instaurer plus tôt.

A l'aide des graphes nous sommes également en mesure de savoir si des espaces de type « réseaux » sont apparus entre les différentes époques. En effet, d'un simple coup d'œil, nous sommes capable de voir qu'entre 1813 et 1972, il y a un réseau qui est apparu d'ouest au Nord-Est de notre zone d'étude de la commune d'Aubigné-Racan, en magenta sur la figure 33 ci-dessous. Cela se traduit par l'augmentation des relations d'adjacences indirectes (en rouge) dans la zone concernée.

Figure 33 Exemple d'utilisation des graphes pour détecter les changements de réseaux

Nous pouvons également faire quelques analyses de l'évolution du parcellaire. Nous pouvons remarquer qu'entre 1850 et 1972-74 de nombreuses parcelles ont été fusionnées, certainement après une opération de remembrement de grande envergure. Cela se traduit par une diminution des relations d'adjacences, nous aurons beaucoup moins d'arêtes pour une même zone, ici représentée en bleu foncé sur la figure 34 ci-dessous, entre les deux époques. Il pourrait être effectué sur ces graphes des analyses quantitatives des relations d'adjacence pour analyser l'évolution du parcellaire dans les différentes feuilles.

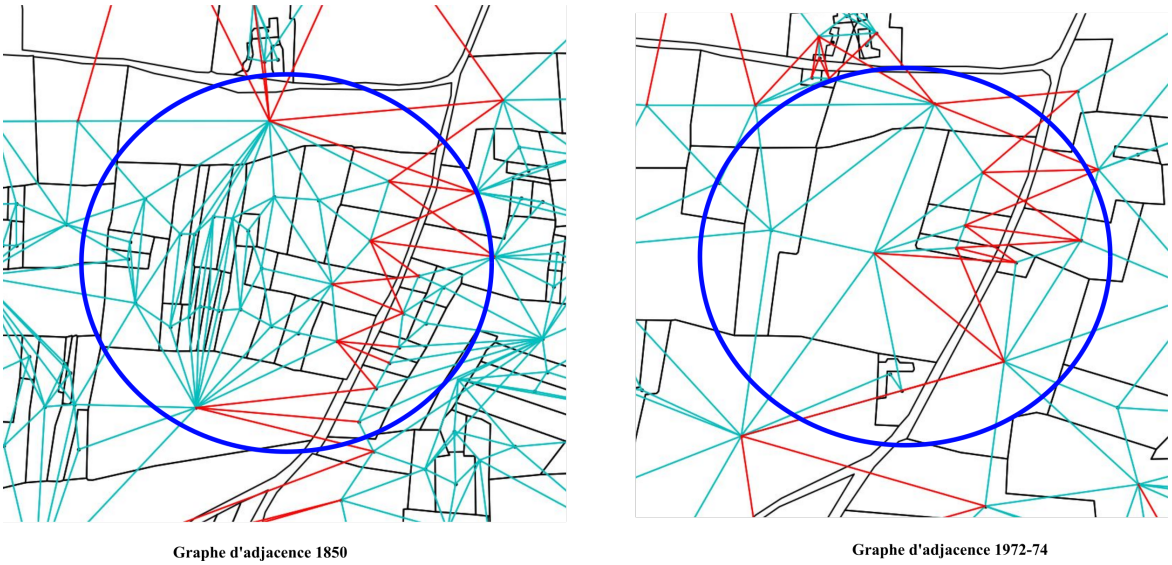


Figure 34 Exemple d'utilisation du graphe d'adjacence pour suivre l'évolution du parcellaire

Les liens entre le mosaïquage et la théorie des graphes sont, dans notre étude, très forts. En effet la génération du graphe d'adjacence directe et indirecte découle directement du mosaïquage des données et de la récupération des données.

Conclusion et perspectives

La chaine de traitement proposée par le laboratoire GeF du CNAM dans le cadre du projet « Le cadastre comme outil d'analyse de l'évolution des territoires » continue de s'étoffer. Après que les étapes de vectorisation et de géoréfencement aient été traitées précédemment (FAHRASMANE (2015) et ODIE (2016)), nous avons donc réalisé l'étape du mosaïquage.

Au cours de ce TFE nous avons développé une chaine de traitement basée pour la majeure partie sur des outils intégrés dans QGIS. Cette chaine de traitement pourra être améliorée par la modélisation des données par un graphe. Elle est très intéressante à utiliser notamment dans l'aide à la détection d'erreur après mosaïquage et pour l'analyse spatio-temporelle avec l'utilisation des graphes temporels. Cette approche pourrait être utilisée prochainement pour l'étape suivante de la chaîne de traitement : la création de base de donnée multi-époque et l'analyse de l'évolution du territoire.

Les résultats obtenus sont en adéquation avec les hypothèses émises en début de TFE, c'est-à-dire que les planches de 1813 offrent un mosaïquage de moins bonne qualité que celles de 1972-74. Cela est en partie dû à l'évolution des techniques de création et de conservation du cadastre entre les différentes époques. Il n'en reste pas moins que si nous souhaitons avoir un résultat plus convenable, il faut qu'en amont du mosaïquage, il y ait eu un nettoyage minutieux de la vectorisation.

L'outil que nous avons proposé n'est pas parfait, il présente encore quelques défauts. Certaines parcelles apparaissent en double malgré la gestion des objets dupliqués tout au long de la chaine de traitement et doivent être corrigées en sortie de chaine. Il y a également l'étape de la triangulation de Delaunay restreinte des espaces non cadastrés, qui n'est pas intégrée dans l'automatisation, et que nous n'avons réussi à produire que par l'outil proposé par le logiciel Open Jump. Il serait intéressant de trouver une manière d'automatiser cette étape et de se passer de l'outil Open.

Nous en venons au sujet concernant la vectorisation. Celle-ci reste à améliorer, il reste des soucis concernant les limites des biens non délimités qui apparaissent en pointillés sur l'image du cadastre et qui en sont pas détectées par la transformée de Hough et l'outil de suivi de chemin de GRASS. De plus, certaines parcelles ne sont pas détectées car elles sont trop petites ou mal corrigées par les outils de GRASS à cause des seuils incertains de certains outils.

Enfin, dans le but d'améliorer la chaine de traitement, il serait intéressant de réaliser différentes chaines de traitement suivant l'ancienneté des planches utilisées. Il ne reste donc plus que l'étape de l'intégration dans la base de données multi-temporel à effectuer pour terminer la chaine de traitement.

Bibliographie

Articles de revues scientifiques et ouvrages :

Bakkouch et al. (2015). Comparaison de méthodes de recalage d'images pour le géoréférencement de clichés anciens. *Revue XYZ n°142 - 1er trimestre 2015,* pp 53-61.

Boison C. (2012). Etude de faisabilité de mise à disposition du plan cadastral napoléonien sur GéoBretagne. Rapport de stage DIPLÔME MASTER SIGAT, Université de Rennes 2, 2012, pp.

Claramunt C. et Libourel T. (2001). A Spatio-Temporal Model for the Manipulation of Lineage Metadata. *Geoinformatica Janvier 2001.*

Clergeot P. et Berteau G. (2008). Du cadastre napoléonien au cadastre en ligne sur Internet. *Revue XYZ n° 119 - 1er trimestre 2009. Textes issus d'une conférence « à deux voix » donnée au CNAM à Paris le 11 Décembre 2008, par M.Gilles Berteau et Pierre Clergeot,* pp 49-59.

Costes B. (2016). Vers la construction d'un référentiel géographique ancien : un modèle de graphe agrégé pour intégrer, qualifier et analyser des réseaux géohistoriques. Géographie. Université Paris-Est, pp.

Del Mondo G. (2011). Un modèle de graphe spatio-temporel pour représenter l'évolution d'entités géographiques. Thèse soutenue le 14 Octobre 2011 pour obtenir le titre de DOCTEUR de l'UNIVERSITÉ de BREST, mention : Informatique – Géomatique.

Fahrasmane M. (2016). Géoréférencement de cadastres anciens vectorisés au moyen de logiciels libres : Application sur les communes de Vaas et Aubigné-Racan (Sarthe). Mémoire de soutenance de DIPLÔME D'INGENIEUR CNAM Spécialité : Géomètre Topographe, ESGT, 2016, pp.

Ferraton A. et Zevort Michel (2017). Jeux de données SIG – Vérification et correction des géométries. Rapport : Centre de Prestations d'Ingénierie Informatique. Département

Follin J.-M., Fahrasmane M., Simonetto E., (2016). An open-source based toolchain for the georeferencing of old cadastral maps. Actes OGRS2016 Open Source Geospatial Research & Education Symposium, Pérouse (Italie), 12-14 octobre 2016.

Gauthiez B. (2008). Lyon en 1824-32 : un plan de la ville sous forme vecteur d'après le cadastre ancien. *Géocarrefour vol. 83 1/2008*. Université de Lyon. pp 57-67.

Hedefalk et al. (2017). Data Descriptor : Spatiotemporal historical datasets at micro-level for geocoded individuals in five Swedish parishes, 1813-1914. *Revue Scientific Data*.

Hedefalk, F. et al. (2017). Spatiotemporal historical datasets at micro-level for geocoded individuals in five Swedish parishes, 1813-1914. *Sci. Data* 4 :170046 doi : 10.1038/sdata.2017.46

Labrousse E. et al., (2015). Evolution du parcellaire cadastral à l'échelle communal : proposition méthodologiques et retour d'expérience à partir des cas des communes d'Aubigné-Racan et de Vaas (Sarthe) de 1813 à nos jours. Laboratoire GeF, document interne.

Le Couédic et al. (2012). Du cadastre ancien au graphe. Les dynamiques spatiales fiscales et modernes. *Revue ArchéoSciences 36.* Presses universitaires de Rennes. pp 71-84.

Martinez-Llario J. (2017) Rule-based topology system for spatial databases to validate complex geographic datasets. Revue *Computer & Geosciences n°103* pp. 122-132

Odie C. (2017). Vectorisation semi-automatique de planches scannées du cadastre ancien. Mémoire de soutenance de DIPLÔME D'INGENIEUR CNAM Spécialité : Géomètre Topographe, ESGT, 2017, pp.

Opérationnel Ouest. Ministère de l'Écologie, de l'Énergie, du Développement durable et de la mer en charge des Technologies vertes et des Négociations sur le climat.

Randell et al. (1992). A spatial logic based on regions and connection. *Principles of Knowledge Representation and Reasoning : Proceedings of the Third International Conference,* pp 165-176, San Mateo, California.

Stell J. (1999). Granulation for Graphs. Departement of Computer Science. Keele University.

Theobald D. (2001) . Topology revisited : representing spatial relations. *Int. J. Geographical information sciences, 2001, vol. 15, no.8,* pp 689-705.

Site internet :

Triangulation de Delaunay. Définition de la triangulation de Delaunay, [en ligne]. Disponible sur : https://fr.wikipedia.org/wiki/Triangulation_de_Delaunay

Outils d'analyse spatiale, Description des outils d'analyse spatiale, [en ligne]. Disponible sur : http://www.spatialanalysisonline.com/HTML/index.html

PySal. Description de la bibliothèque PySal, [en ligne]. Disponible sur : http://pysal.readthedocs.io/en/latest/users/tutorials/weights.html#spatial-weights

Table des figures

Figure 1 Principe d'enchaînement des étapes de la chaine méthodologique du Laboratoire GeF, d'après Fahrasmane (2016) 8

Figure 2 Extraits des cadastre de 1813, 1850, 1972-74 (de gauche à droite) 11

Figure 3 Différents niveaux d'échelle du cadastre, d'après Faharsmane (2016) 11

Figure 4 Problème de multi-ligne provoqué par l'épaisseur des lignes et les liserais sur le format raster..... 13

Figure 5 Choix d'un seuil trop important pour la fonction snap : avant/après..... 15

Figure 6 Croisements en "V" aux intersections de parcelles 15

Figure 7 Différence de repères pour le vecteur et le raster 15

Figure 8 Problème de polygone inclut dans un autre lié à la labellisation 16

Figure 9 Relations topologiques du RCC8, d'après Randell et al. (1992) 17

Figure 10 Graphe (en bleu) et graphe dual (en rouge)..... 20

Figure 11 Triangulation de Delaunay, d'après Hervé Lehning..... 22

Figure 12 Analogie avec un élastique. La ligne bleue formant l'enveloppe convexe, d'après Wikipédia l'encyclopédie libre 23

Figure 13 Illustration de l'union..... 23

Figure 14 Illustration de l'intersection 23

Figure 15 Exemple de fusion de tables attributaires..... 24

Figure 16 Différents cas de corrections topologiques à apporter 26

Figure 17 Problème de liaison entre parcelles 27

Figure 18 Planches géoréférencées non corrigées 29

Figure 19 Planches corrigées des chevauchements 31

Figure 20 Planches corrigées des chevauchements et des trous 33

Figure 21 Graphe planaire du parcellaire..... 34

Figure 22 Graphe dual (adjacence) du graphe planaire 34

Figure 23 Exemple de graphe temporel, d'après G. Del Mondo (2011) 35

Figure 24 Graphe d'adjacence de la planche 013 avant/après correction 37

Figure 25 Problème de densification des sommets..... 40

Figure 26 Résolution du problème d'adjacences indirectes 40

Figure 27 Extrait du géoréférencement des planches de 1813 43

Figure 28 Planches corrigées des chauvauchement (à gauche) planches partiellements corrigées des trous (à droite) 44

Figure 29 Problème de parcelle dupliquée (ici en jaune) 45

Figure 30 Mosaïquage 1972-74 avant/après 45

Figure 31 Exemple de résultat pour le graphe d'adjacence d'une planche de 1813 46

Figure 32 Exemple d'adjacence manquée dans le programme Python (ici en magenta)..... 47

Figure 33 Exemple d'utilisation des graphes pour détecter les changements de réseaux 48

Figure 34 Exemple d'utilisation du graphe d'adjacence pour suivre l'évolution du parcellaire49

Liste des tableaux

Tableau 1 Propriétés des images scannées de la commune Aubigné-Racan, d'après Fahrasmane (2016)..... 12

Tableau 2 Données utilisées suivant les différentes époques..... 12

Tableau 3 Liste des logiciels ayant des aides à la correction topologique (d'après J.Martinez-Lario et al. 2017)..... 19

Tableau 4 Relations topologiques possibles entre différentes entités présentes sur une planche de cadastre..... 25

Tableau 5 Exemple de matrice d'adjacence 37

Annexes

Annexe 1 : Géoréférencement des planches de 1813

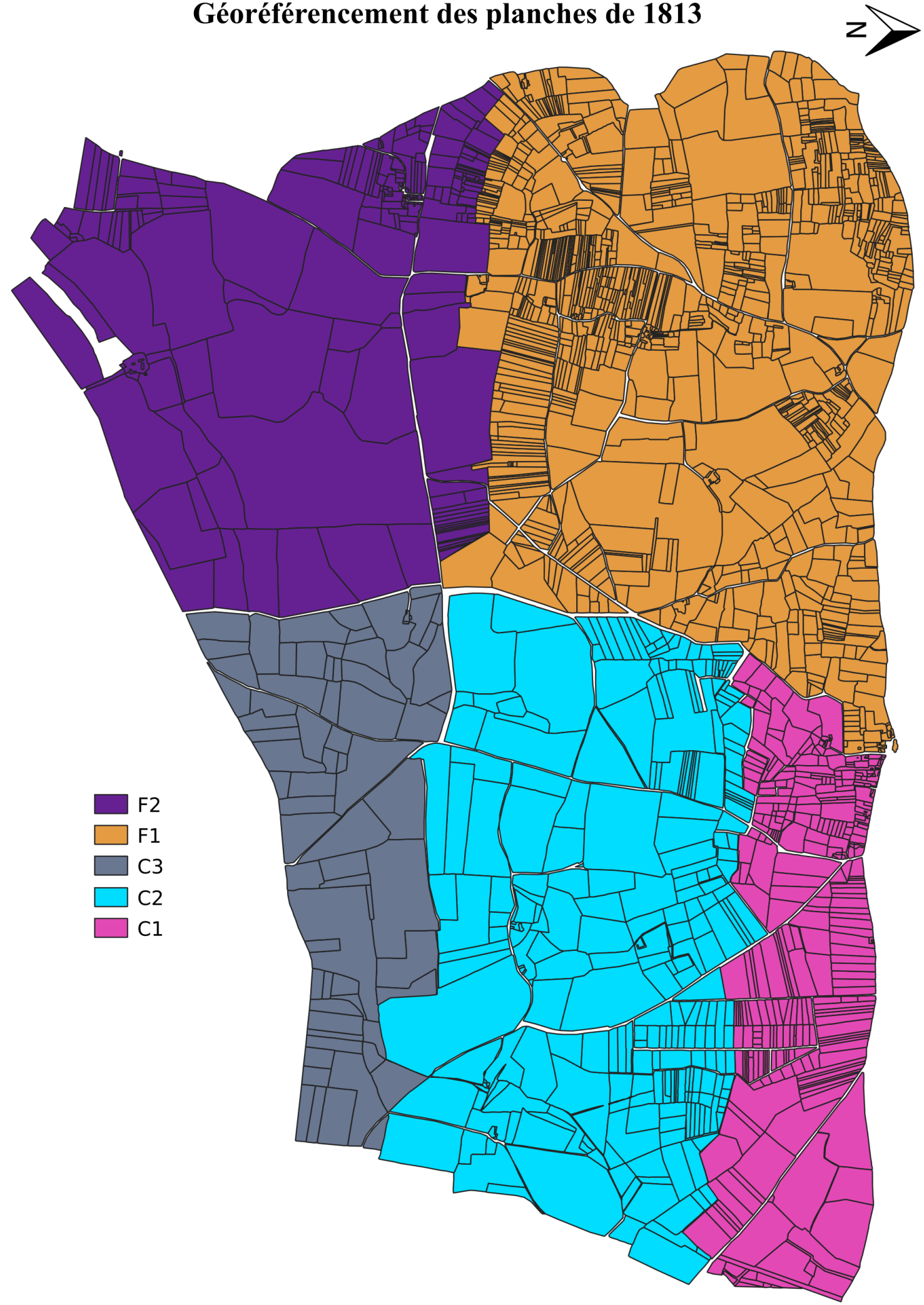
Annexe 2 : Mosaïquage des planches de 1850

Annexe 3 : Mosaïquage des planches de 1972-74

Annexe 4 : Graphe de l'année 1850

Annexe 5 : Graphe de l'année 1972-74

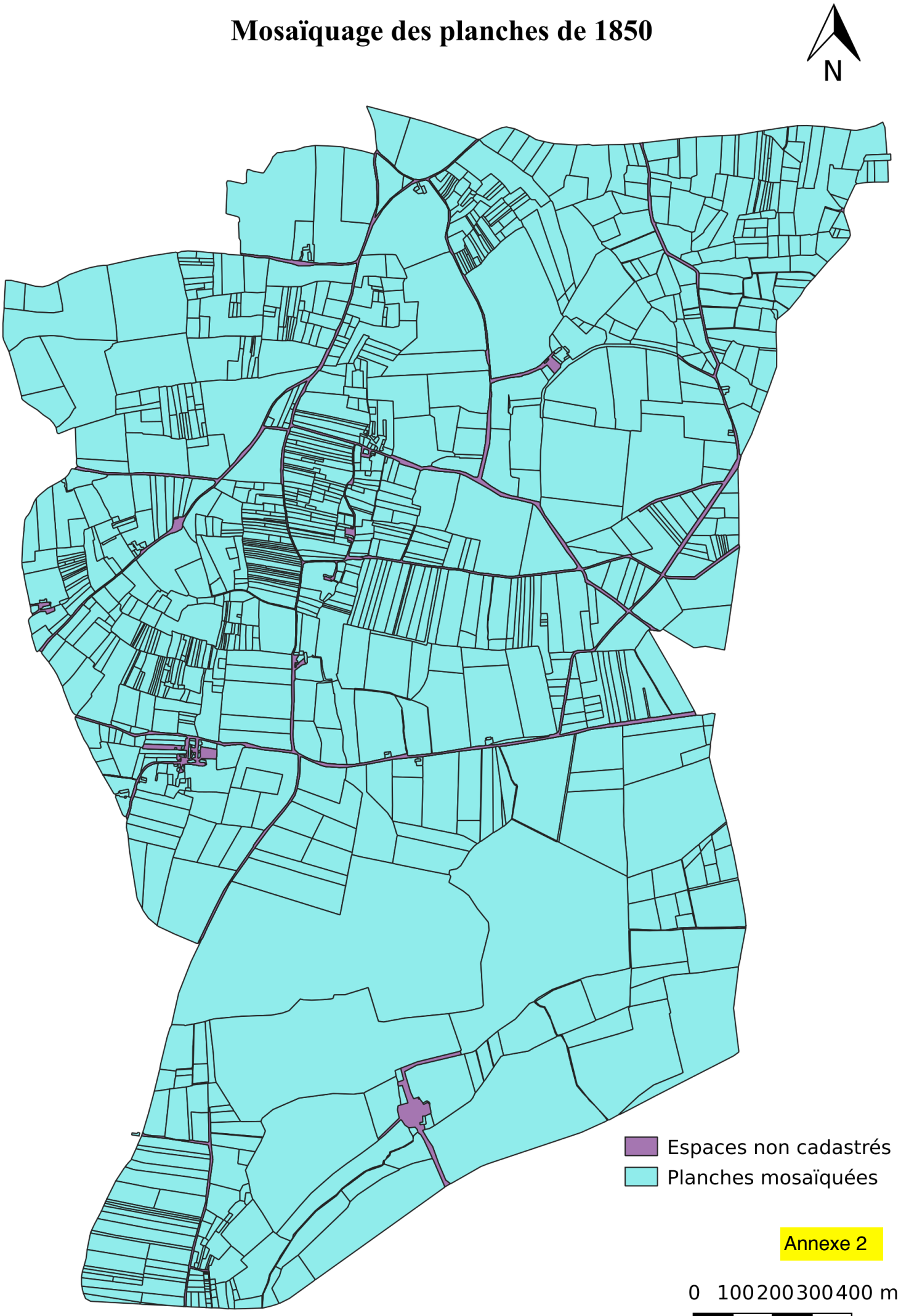
Géoréférencement des planches de 1813



Annexe 1

Source: Issu de la vectorisation et du géoréférencement de cinq planches de 1813

Mosaïquage des planches de 1850



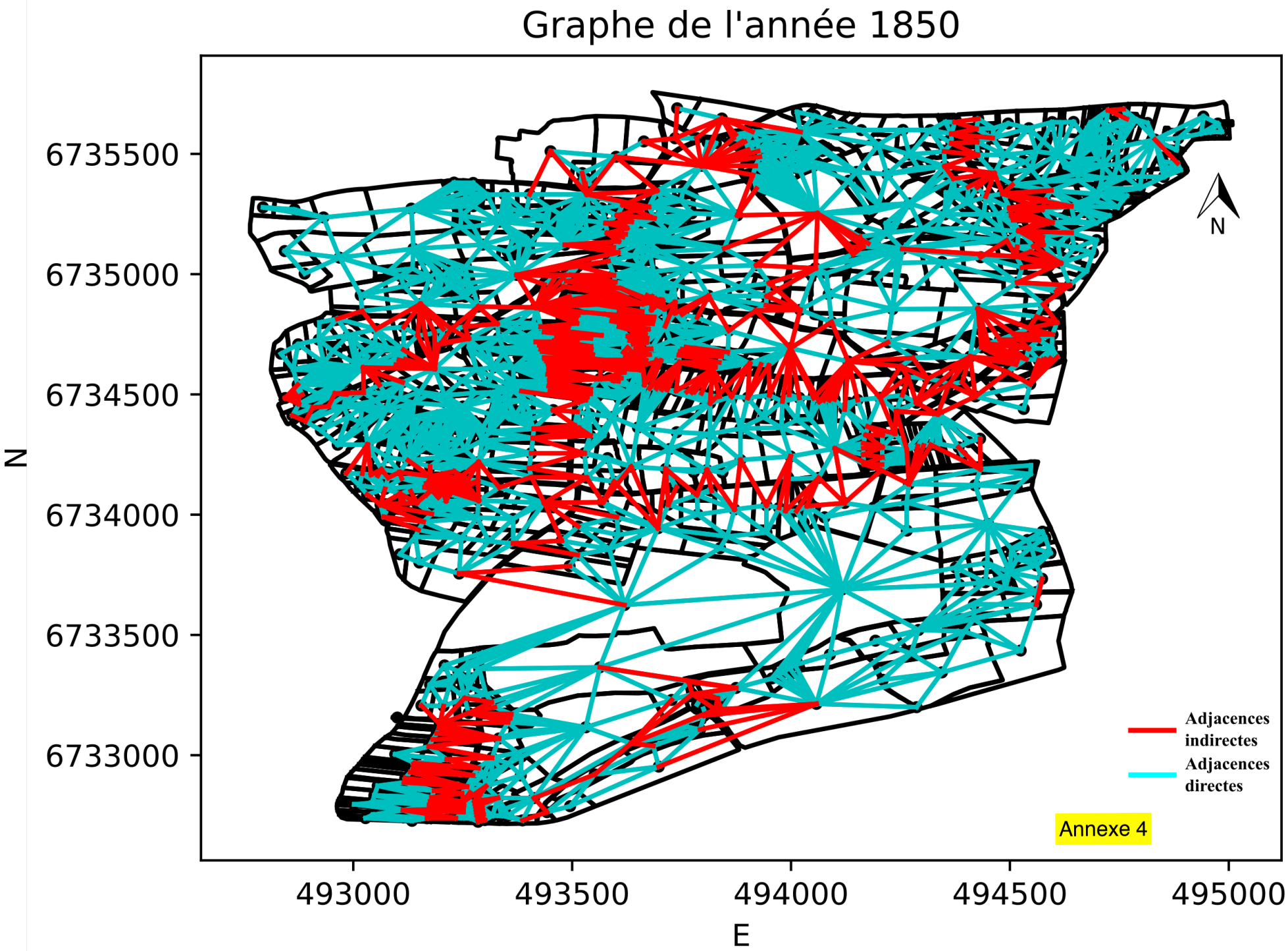
Source: Issu de la chaine de traitement de mosaïquage QGIS

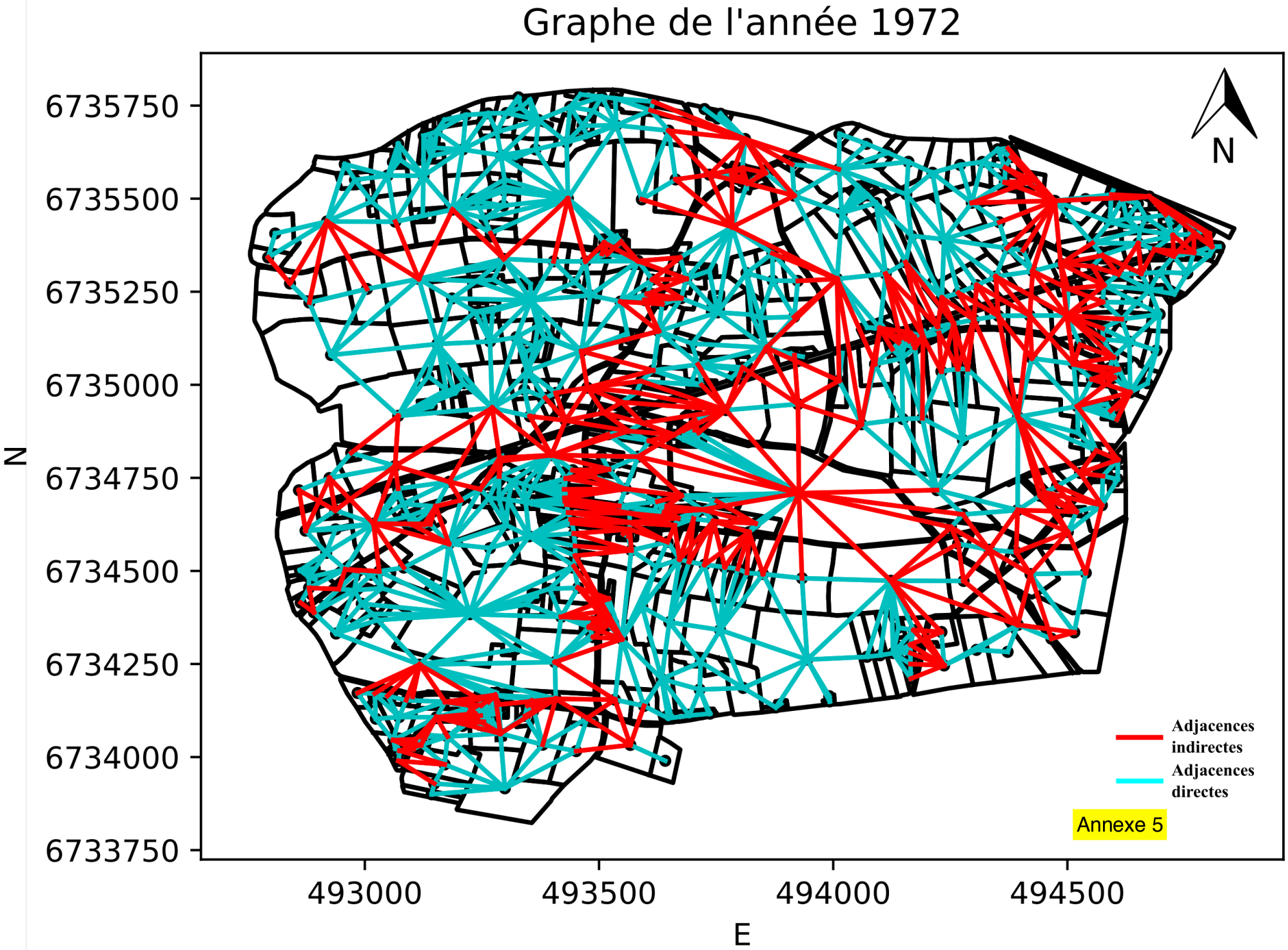
Mosaïquage des planches de 1972-74

Annexe 3

0 100 200 300 400 m

Source: Issu de la chaine de traitement de mosaïquage QGIS


Graphe de l'année 1850
6735500
6735000
6734500
6734000
6733500
6733000
N
493000
493500
494000
494500
495000
E
N
Adjacences indirectes
Adjacences directes
Annexe 4


Graphe de l'année 1972
6735750
6735500
6735250
6735000
6734750
6734500
6734250
6734000
6733750
N
493000
493500
494000
494500
E
N
Adjacences indirectes
Adjacences directes
Annexe 5

Résumé

Depuis quelques années, nous constatons un engouement pour l'étude des documents anciens. Le cadastre napoléonien constitue un outil d'analyse temporel de l'évolution des territoires à différentes échelles. Afin d'avoir une analyse plus poussée que par l'étude des documents papier du cadastre, il est nécessaire de les vectoriser et de les géoréférencer. Seulement, le géoréférencement induit de nombreuses erreurs entre les différentes données vectorisées, telles que des trous ou des chevauchements des parcelles. Des outils de correction existent sur des logiciels propriétaire, mais la chaine de traitement proposée par le Laboratoire Géomatique et Foncier s'appuie sur des outils open source. Dans ce TFE, nous allons nous intéresser à la correction de ces erreurs au moyen de deux méthodes : un mosaïquage basé sur une chaine d'outils proposés par le logiciel QGIS, et une approche basée sur la théorie des graphes et des relations de voisinage afin de proposer un guidage du mosaïquage.

Mots clé : Cadastre Ancien, Mosaïquage, Théorie des Graphes, Automatisation, SIG, GRASS, Python, Logiciel Open Source, Relations d'Adjacence, Post-Traitement.

Abstract

In recent years, we note a craze for the study of old documents. The Napoleonic cadastre is a tool for temporal analysis of the evolution of territories at different scales. In order to have a more detailed analysis than by studying the paper documents of the cadastre, it is necessary to vectorize and georeference them. However, georeferencing induces many errors between the different vectorized data, such as holes or overlapping between plots. Correction tools exist on proprietary software, but the processing chain proposed by the Laboratoire Géomatique et Foncier is based on open source tools. In this TFE, we will focus on the correction of these errors by two methods: tiling based on a toolchain proposed by QGIS software, and an approach based on graph theory and neighborhood relationships to provide tiling guidance.

Keywords : Historical Cadastral Map, Tiling, Graph Theory, Automation, GIS, GRASS, Python, Open Source Software, Adjacency Relation, Post Treatment